

TARTU ÜLIKOOL
MATEMAATIKA-INFORMAATIKATEADUSKOND
Arvutiteaduse Instituut
Informaatika eriala

Risto Soonvald

Reaalaja kell LEGO MINDSTORMS NXT jaoks
Bakalaureusetöö (6 EAP)

Juhendaja: lektor Anne Villems
Kaasjuhendaja: spetsialist Taavi Duvin

Autor:““ mai 2011
Juhendaja:““ mai 2011
Juhendaja:““ mai 2011

Lubada kaitsmisele
Professor:““ mai 2011

TARTU 2011

Sisukord

Sissejuhatus	3
1. Aeg ja selle mõõtmine	5
1.1. Aja määramine	5
1.2. Ajavööndid.....	10
1.3. Suveaeg	10
2. Reaalaja kell RTC-Nx-v2	12
2.1 Reaalaja kella lühitutvustus	12
2.2 Reaalaja kella tööpõhimõte.....	13
2.3 Oluline jälgida.....	19
2.4 Reaalaja kella programmeerimine	20
2.4.1 Reaalaja kella programmeerimisploki lisamine LEGO MINDSTORMS Education NXT keskkonda	21
2.4.2 RTC-Nx ploki kuvaseaded	24
2.4.3 RTC-Nx ploki konfigureerimine	25
2.4.4 RTC-Nx ploki infotulp	26
3. Ülesanded reaalaja kella RTC-Nx jaoks.....	29
3.1 Ülesanne 1 - Anduri kasutamine kellana	29
3.2 Ülesanne 2 - Andmete kogumine.....	32
3.3 Ülesanne 3 - Äratuskell.....	34
3.4 Ülesanne 4 - Eestikeelne väljund.....	36
3.5 Ülesanne 5 - Punamütsikese mure	38
3.6 Ülesanne 6 - Vilkuvad tuled	40
3.7 Ülesanne 7 - Seinakell	41
3.8 Ülesanne 8 - Binaarkell.....	43
Kokkuvõte	46
Abstract.....	47
Viited	48
Lisad	52
Lisa 1. Andmeedastus I ² C siinil.....	52
Lisa 2. Reaalaja kella registrid.....	53
Lisa 3. Reaalaja kella infotulp	54
Lisa 4. CD ülesannete lahendusfailidega	56

Sissejuhatus

Reaalained ei ole Eesti koolides eriti populaarsed. Õpilased peavad reaalainete õpetamist tuimaks ja igavaks ning nende õppimist liialt raskeks. Sellist mõtteviisi tuleb muuta ning üheks väga lihtsaks võimaluseks on koolidele robotikakursuste tutvustamine. 2007. aastal alustati Eesti koolides Kooliroboti projekti, mille eesmärgiks on suurendada kooliõpilaste seas huvi reaalainete vastu ning edendada seeläbi inseneriteadust [1].

Tehnoloogiasektor on üks kiiremini arenevaid sektoreid ning nii Eestis, kui ka mujal maailmas on suur vajadus inseneride, arendajate ja lihtsalt tehnoloogiahuviliste vastu. Paljud arvavad, et uuendustega pidevalt kaasas käimine on võimatu. Sellise negatiivse arvamuse mahasurumiseks tuleb ümber kujundada olemasolevaid õppeprogramme, et tekitada õpilastes huvi tehnoloogia vastu. Koos sellega suureneb võimalus, et hilisemad koolilõpetajad hakkavad tulevikus selles sektoris töötama.

Uute tehnoloogiatega kursis olemine on tänases info- ja kommunikatsioonitehnoloogia valdkonnas väga oluline ning esmase huvi äratamiseks pole paremat meetodit, kui kasutada LEGO MINDSTORMS NXT roboteid ning sellega ühilduvaid andureid.

Robotikakursustel kasutatavaid LEGO MINDSTORMS NXT roboteid on pärast põgusat juhendamist võimelised programmeerima ka kõige tehnoloogiakaugemad inimesed. Lisaks lihtsale programmeerimisele on LEGO MINDSTORMS NXT robotitega võimalik lahendada ka väga keerulisi ülesandeid. Seepärast sobivad robotikatunnid erineva tasemega õppuritele. Praegusel hetkel on mitmetes koolides võimalik võtta kursuseid juba alates algklassidest ning vanuse ülempiiri pole seatud. Seetõttu on ka LEGO MINDSTORMS NXT stardikomplekti karbile märgitud vanusepiiranguks 8+, sest see on piisavalt lihtne nii algklassiõpilastele kui ka parajalt keeruline kogenud robotiehitajatele. LEGO robotid teeb õpilastele atraktiivseks just see, et kohe on võimalik näha tulemusi – robot kas teeb häält, liigub või suhtleb kasutajaga mõnel kolmandal viisil. LEGO MINDSTORMS NXT roboti stardikomplekti kuuluvad roboti juhtaju ning mitmed huvitavad andurid ning mootorid. Kui nendest anduritest jääb väheseks, siis on võimalik andureid juurde tellida, sest vastavaid tootjaid on mitmeid. Probleemiks on see, et uutel anduritel puuduvad eestikeelsed kasutusjuhendid ja ülevaated ning robotikakursuste läbiviijatel puudub vastav aeg ja ressurss uute anduritega tutvumiseks. Uutest anduritest

tehakse lõputööd kirjutavate tudengite poolt põhjalikud ülevaated, mida hakatakse kasutama hilisemas õppetöös.

Antud bakalaureusetöö keskendub ühele konkreetsele LEGO MINDSTORMS NXT robotile mõeldud lisaandurile - reaallaja kellale RTC-Nx-v2, mille tootjaks on firma Mindsensors. Andurist tehakse põhjalik ülevaade, demonstreeritakse selle võimalusi ning pakutakse lahendamiseks ülesandeid. Ülesanded on loodud erinevate raskusastmetega, et rahuldada erineva tasemega õpilaste huvisid. Anduri võimalusi kirjeldatakse sellisel moel, et ka inimene, kes pole vastava valdkonnaga kunagi kokku puutunud, suudaks lugeda asja huviga, sellest aru saada ning on pärast mõningast juhendamist suuteline ise programme koostama. Lisaks reaallaja kella anduri kirjeldamisele näidatakse võimalusi, kuidas seda andurit kasutada koostöös teiste anduritega.

Bakalaureusetöö eesmärgiks on anda eestikeelne ülevaade vastavast andurist ning selle võimalustest, tekitada õpilastes täiendavat huvi robotika vastu ning arendada Kooliroboti projekti koolides, kuid materjal on kättesaadav ka lihtsalt asjahuvilistele. Lõputöö koosneb kolmest peatükist, kus kirjeldatakse vastava anduri tööpõhimõtet, kasutamist ning programmeerimist.

Esimene peatükk annab üldise ülevaate aja mõõtmise ajaloost ning vajalikkusest. Teises peatükis kirjeldatakse lähemalt reaallaja kella RTC-Nx-v2 olemust ning võimalusi. Lisaks sellele antakse ülevaade anduri kasutamise- ning programmeerimisvõimaluste kohta. Kolmas peatükk koosneb spetsiaalsetest andurile mõeldud ülesannetest. Peale selle antakse ka ülevaade võimalikest lahenduskäikudest. Ülesanded on mõeldud kasutamiseks LEGO MINDSTORMS NXT robotil ning nende loomiseks on kasutatud graafilist NXT-G programmeerimiskeskonda.

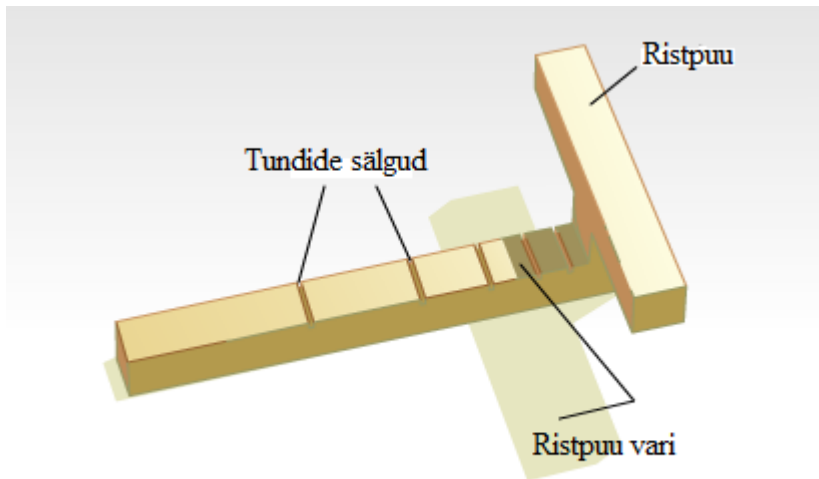
1. Aeg ja selle mõõtmine

Enne reaalaraja kella tutvustamise juurde asumist antakse ülevaade inimkonna pingutustest kellaaja määramisel läbi ajaloo. Alustuseks kirjeldatakse esimesi, väikese täpsusega meetodeid. Peatüki lõpuks jõutakse tänapäevaste tõeliselt täpsete ajahetki fikseerivate meetoditeni.

1.1. Aja määramine

Inimene on aastatuhandeid püüdnud aega määrata, saamaks teada, millal jaht lõpetada või millal Päike loojub. Looduses esinevate tsüklite ja mustrite jälgimine ning nende tõlgendamine võimaldab aega suhteliselt õigesti hinnata. Inimesed on jälginud Päikese liikumist ja avastanud, kuidas määrata aega päikeseloojanguni. Selleks tuleb pilk kaugusesse suunata, käsi ette sirutada, käelaba paralleelselt kehaga panna ja nimetissõrm täpselt Päikese alla asetada. Vastavalt sellele, kui palju käelaba laiuseid ribasid Päikese ja maapinna (horisondi) vahele mahub, nii mitu tundi on aega päikeseloojanguni. Tänapäeval kasutavad sellist meetodit vähesed, kuid aastasadu tagasi võis selle kasutamine olla elu ja surma küsimus. Sellegipoolest on inimene alles hiljaaegu leidnud võimaluse tõeliselt täpselt aega kindlaks teha.

Egiptlased olid esimesed inimesed, kes võtsid kasutusele 24 tunni pikkuse ööpäeva mõiste. Öö oli jagatud 12 tunniks, päev 10 tunniks ning 1 tund oli videvikuaeg enne koitu ning pärast eha. Öösel sai aega määrata tähtede asukoha järgi [2]. Päeval kasutati aga varjukella (inglise keeles *shadow clock*) (Joonis 1), mis oli kasutusel juba 8. sajandil eKr. Varjukell on piisavalt pisike ja mugav, et seda kaasas kanda, sest mahub taskusse. Hommikul peab kella pikem tunnisälkudega märgistatud osa olema suunatud läände ning pärastlõunal itta. Vastavalt Päikese liikumisele tekib varjukella ristpuu vari kella sälkudega küljele. Selleks, et varjukella ka pärastlõunal kasutada saaks, tuleb selle pikem osa suunata itta [3]. Tegemist ei ole kõige lihtsamini kasutatava vahendiga, sest kasutaja peab teadma täpselt ilmakaarte asetsemist vastavalt oma asukohale ning pilves ilmaga ei tööta vahend üldse [4]. Selline mõõtmisvahend sõltub liialt palju välistest teguritest.



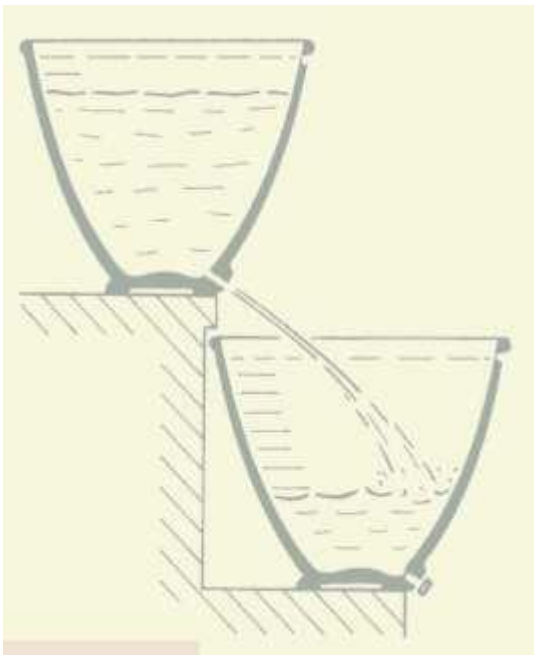
Joonis 1. Egiptlaste poolt kasutatud varjukell [3].

Varjukella võib nimetada päikesekella eelkäijaks. Lisaks egiptlastele püüdsid ka teised rahvad, nagu näiteks hiinlased, babüloomlased, kreeklased ja roomlased, leida erinevaid vahendeid kellaaja määramiseks. Muistsed rahvad märkasid, et päikese trajektoor on suvel ja talvel erinev. Seetõttu ei näidanud algsed päikesekellad aastaringselt õiget aega. Pärast mitmeid ebaõnnestumisi avastati, et kui panna päikesekella (inglise keeles *sundial*) (Joonis 2) vertikaalne osuti kindla nurga all seisma, siis töötab see terve aasta [2]. Lisaks sellele tuleb osuti suunata põhjapoolusele. Näiteks vertikaalselt asetseva päikesekella numbrilaud peab moodustama osutiga $90^\circ - \alpha$ kraadise nurga, kus α võrdub kella asukoha geograafilise laiusga, samas horisontaalselt paikneva päikesekella numbrilaud moodustab osutiga α kraadise nurga. Horisontaalne päikesekell muutub kasutuks ekvatoriaalsetel aladel, sest sellisel juhul $\alpha = 0^\circ$ ja see tähendab, et kella numbrilauale ei saa varju tekkida [5].



Joonis 2. Tüüpiline horisontaalne päikesekell [6].

Öösel oli kellaaja määramine tunduvalt keerulisem, sest tuli jälgida tähti. Selleks, et Egiptuse kuningas Amenophis I ei peaks kellaaja vaatamiseks öösel oma voodist välja ronima, leiutati veekell (kreeka keeles *clepsydra*), mis kreeka keeles tähendab vett varastama (Joonis 3). Veekell on palju usaldusväärsem, kuna ei sõltu niivõrd muutuvatest faktoritest. Vesi tilgub seadme ülemisest anumast alumisse anumasse, millele on märgitud tundide jooned. Kui vesi jõuab vastava jooneni, on järjekordne tund möödunud. Paraku ei olnud veekell väga populaarne, kuna tilkumise kiirus sõltus paljuski vee temperatuurist ja puhtusest [2, 7].



Joonis 3. Kahe anumaga veekell [8].

Veekella puuduseid arvesse võttes ja neid elimineerides leiutati liivakell, (inglise keeles *hourglass*) (Joonis 4) mida kasutatakse edukalt ka tänapäeval. Liivakell on geniaalne leiutis, kuna liiv asub kinnises konteineris ning seda ei mõjuta välised tegurid [2]. Liivakellaga saab mõõta aega mõnest minutist kuni mitme tunnini. Liiv voolab ülemisest anumast alumisse ning kui anum tühjaks saab, keeratakse kell ümber. Tänapäeval kasutatakse liivakella näiteks lauamängudes mängukorra aja mõõtmiseks või köögis muna keetmisel.



Joonis 4. Tüüpiline liivakell [9].

1386. aastal leiutati esimene mehaaniline kell. Kella all ripub raskus, mis liigub aeglaselt alla, liigutades hammasrattaid. Sellised kellad pidid asuma suhteliselt kõrgetes tornides, et need töotaksid mõistlikult kaua. Selle aja inimeste arvates olid sellised kellad vägagi täpsed, sest erinesid tegelikust kellaajast ainult paari tunni võrra [2]. Tõeliselt täpsete kellade loomisele andis suure panuse itaallasest füüsik, matemaatik, astronoom ning filosoof Galileo Galilei, kes 1581. aastal 17-aastaselt jälgis igavusest Pisa katedraali lühtri võnkumist ning hakkas pendleid uurima. 1602. aastal avastas Galileo pendliefekti, mis tähendab et pendlil kulub ühe täiskaare tegemiseks alati sama pikk aeg, olenemata kaare pikkusest (amplituudist) [2, 10, 11]. Galileo mõtteid kasutades valmistas 1656. aastal Christiaan Huygens esimese pendelkella [12]. Pendelkell oli küll täpne, kuid ei toiminud merel, kus lainetus kella pendli rütmi häiris. Meremeestel oli oma asukoha määramiseks vaja teada täpset aega. Põhjast lõunasse liikudes jälgiti Põhjjanaela. Raskusi asukoha määramisel valmistas just idast läände liikumine. Kui neli Briti laeva 1707. aastal Sitsiilia saarte lähistel karidele sõitsid, korraldas Briti valitsus konkursi loomaks merele sobivat kella [2]. Kella abil pidi saama reisida Suurbritanniast Kariibi meres asuvate Briti kolooniasse kuuluvatele saartele poole kraadise täpsusega, mis ajaliselt tähendab umbes kahte minutit. Vaevatasuks pakuti £20 000. Mitmed katsetused ja ebaõnnestumised viisid selleni, et 1761. aastaks sai John Harrison sobiva kella valmis. See oli tema neljas võistluse tarbeks disainitud kell, mis oli algselt mõeldud ainult isiklikuks kasutamiseks. Kella esmasel katsetamisel erines see 31 ööpäeva pikkuse merereisi järel tegelikust ajast vaid 5,1 sekundit. Kella nimeks sai H4 (Joonis 5) ja seda võis nimetada suureks taskukellaks. Selle läbimõõt oli 13 cm ja kaal 1,45 kg [13–15].



Joonis 5. John Harrisoni poolt tehtud merele sobiv kronomeeter kell H4. Esimene kell, mis vastas Briti valitsuse poolt määratud nõuetele [16].

Tänapäeva kellad on läinud väga täpseks ning suudavad ilma sekkumiseta töötada aastaid. 20. sajandil avastasid teadlased, et kvartsikristallidega on võimalik aega mõõta aasta jooksul $\frac{1}{500}$ sekundi täpsusega. 1949. aastal ehitati esimene aatomkell, kuid mis polnud eriti akuraatne. 1955. aastal loodi täpne aatomkell (Joonis 6), mis 300 miljoni aasta jooksul erineb tegelikust ajanäidust vaid 1 sekundi võrra [2]. Bakalaureusetöös kirjeldatavas reaalaja kellas kasutatakse aja määramiseks välist kvartsikristalli. Kellaaja täpsust võib mõjutada temperatuuri või elektrisignaali kõikumine.



Joonis 6. Esimene täpne aatomkell aastast 1955 [17].

Järgnevalt kirjeldatakse, kuidas Maa pöörlemine mõjutab kellaaja määramist ning kuidas on maakera ajavööndite vahel ära jagatud.

1.2. Ajavööndid

Maakera pöörleb ümber oma telje, mis mõjutab olukorda, kus on vastaval hetkel öö ja kus päev. Selleks, et maailmas valitseks ühtne arusaam kellaajast, on Maa jagatud 24 ajavööndi vahel. Ajavööndid on määratud pikkuskraadidena. Maa on ümara kujuga ja seega on ühe vööndi laius $\frac{360^\circ}{24} = 15^\circ$. Ajavööndeid hakatakse lugema nullmeridiaanist, mida tuntakse ka Greenwichi meridiaanina ($7,5^\circ$ nullmeridiaanist itta ning $7,5^\circ$ läände), mis asub Londoni külje all. Greenwichi meridiaan on tähtis sellepärast, et selle koha pikkuskraadiks on 0° ja sellest meridiaanist hakatakse vasakule ning paremale ajavööndeid määrama. Selleks, et riigiti ajaarvamist lihtsustada, on kokkuleppeliselt võetud ajavööndite piirideks riigi- või halduspiirid (muidu oleks näiteks osa Hiiumaast ja Saaremaast teises ajavööndis) [18–22].

Kuna Maa trajektoor ümber Päikese on ellipsi kujuline, siis on tegelik ööpäeva pikkus kuni 7,9 sekundit üle või alla 24 tunni. Maa liigub kiiremini, kui on Päikesele lähemal ning aeglasemalt, kui asub kaugemal. Aasta peale kokku tuleb keskmiseks päeva pikkuseks siiski ligikaudu 24 tundi [23, 24]. Greenwichi meridiaanil päikese järgi mõõdetavat aega nimetatakse GMT (inglise keeles *Greenwich Mean Time*) ajaks. Kuni 1972. aastani keerati kõik maailma kellad vastavalt GMT standardile. Hiljem võeti kasutusel UTC (inglise keeles *Coordinated Universal Time*) standard, mis on suhteliselt sarnane GMT standardile. Üheks erinevuseks on see, et ebaregulaarsete intervallide vahel lisatakse UTC ajale lisasekund. See on tingitud sellest, et Maa pöörlemiskiirus on aeglustumas [20, 25]. Aeglustumise peamiseks põhjuseks on Maa ja Kuu gravitatsiooniline vastastikmõju [26]. 1972. aastast on UTC ajale lisatud praegusel hetkel 24 sekundit [27, 28]. Selleks, et üle maailma oleks ühtlane kellaeg, kasutatakse aja määramiseks UTC standardit. Eesti asub UTC+02:00 vööndis.

1.3. Suveaeg

Suveaeg tähendab vastavalt vööndiajale kellaaja nihutamist ühe tunni võrra edasi. Euroopas minnakse üle suveajale märtsi viimase pühapäeva öösel kell 2:00. Tagasi keeratakse oktoobri viimasel pühapäeval kell 3:00. Suveaega kasutatakse loodusliku valguse efektiivsemaks kasutamiseks ja seeläbi elektrienergia kokkuhoiuks. Samas on

suveaja kasutamise üle palju vaieldud, sest suveaja kasutamine tekitab probleeme näiteks loomakasvatustes, seadmete haldamises, samuti kasvab õhtupäikese tõttu liiklussurmade arv. Suveajale minek võib tekitada probleeme arvutites ja nende poolt juhitud seadmetes. Üldiselt saab tarkvara kellaaja muutusega ise hakkama, kuid võivad tekkida mitmesugused muud komplikatsioonid. Paljud riigid ei kasuta suveaega üldse või kasutavad ainult osaliselt. Hetkel kasutavad suveaega Kanada, USA, Mehhiko, Euroopa riigid, osa Austraaliast, Uus-Meremaa, Paraguai, Tšiili ja Brasiilia [29, 30].

Esimeses peatükis heideti pilk kella ajaloole, kirjeldati ajavööndite olemust ning päevavalguse kasutamise olulisust. Järgnevas peatükis tutvustatakse reaalaaja kella LEGO MINDSTORMS NXT jaoks.

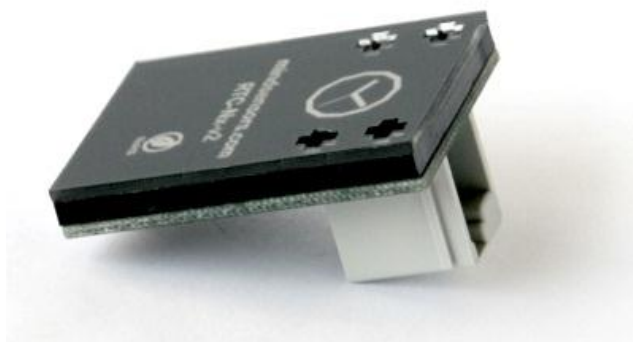
2. Reaalaja kell RTC-Nx-v2

LEGO MINDSTORMS NXT jaoks on toodetud mitmeid erinevaid lisaandureid. Üheks selliseks anduriks on reaalaja kell. Reaalaja kell on selline kell, mis mõõdab aega ka siis, kui süsteem on välja lülitatud. Käesolev peatükk keskendub LEGO MINDSTORMS NXT tarbeks valmistatud anduri reaalaja kell (RTC-Nx-v2) tutvustamisele ning annab sellest pikema ülevaate.

2.1 Reaalaja kella lühitutvustus

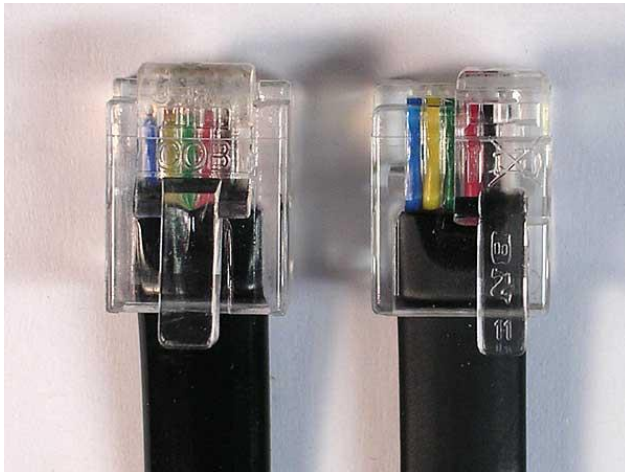
Reaalaja kell (RTC-Nx-v2) on väikese energiatarbimisega kell-kalender [31, 32]. Kella patarei kestab kuni 10 aastat ning kell suudab aega määrata aastasel skaalal kuni sekundi täpsusega. Kell-kalender suudab näidata sekundeid, minuteid, tunde, kuupäeva, kuud, nädalapäeva ning aastat. Juhul kui kuus on vähem kui 31 päeva, kohandatakse see automaatselt vastava kuu päevade arvuga. Lisaks sellele suudab ta arvestada liigaastatega aastani 2100. Kell töötab nii 24 kui ka 12 tunni formaadis koos AM (*ante meridiem*) ja PM (*post meridiem*) tähistusega. Kui kell on ühendatud LEGO MINDSTORMS NXT seadmega kasutab ta sealt saadavat voolu. Voolukatkestuse puhul lülitub kell ise ümber patarei toitele. Informatsiooni salvestamiseks on kellal kasutada 56 baiti NV SRAM (inglise keeles *non-volatile static random access memory*) mälu. Tegemist on staatilise muutmäluga mis lubab andmeid kirjutada ja lugeda ning kus andmed säilivad ka pärast toite väljalülitamist. Mälu hoitakse kellaaajaga seonduvaid andmeid.

Reaalaja kell DS1307 (Joonis 7) on ise üpris väikene, kaaludes vaid 14 grammi. Kella võib hoida temperatuurivahemikus -40° kuni $+85^{\circ}$. Käesolevas töös kirjeldatav kell on mõeldud tavakasutamiseks ning kannatab temperatuurivahemikku 0° kuni $+70^{\circ}$.



Joonis 7. Reaalaja kell RTC-Nx-v2. Mõõtmed (laius x pikkus x kõrgus): 32,6 mm x 42,5 mm x 20,75 mm [33].

Kella ühendamiseks arvutiga tuleb kell ühendada NXT esimesse porti (soovi korral mõnda teise) ning NXT ühendada USB kaabli abil arvutiga. Kella ei saa ühendada otse arvutiga kuna RTC-Nx-v2 omab ainult RJ 12 kaablile (Joonis 8) sobivat pesa. Tavapäraselt asub RJ 12 kaabli kinnitusklamber keskel, kuid NXT puhul on see ääres. See on tüüpiline LEGO MINDSTORMS NXT robotikomplektiga kaasas olev ühenduskaabel lisaseadmete ühendamiseks juhtajuga.



Joonis 8. RJ 12 kaabel. Vasakul on tavapärase kinnitusklambriga ja paremal NXT jaoks mõeldud kaabel [34].

Järgnevalt kirjeldatakse lähemalt reaalaaja kella tööpõhimõtet.

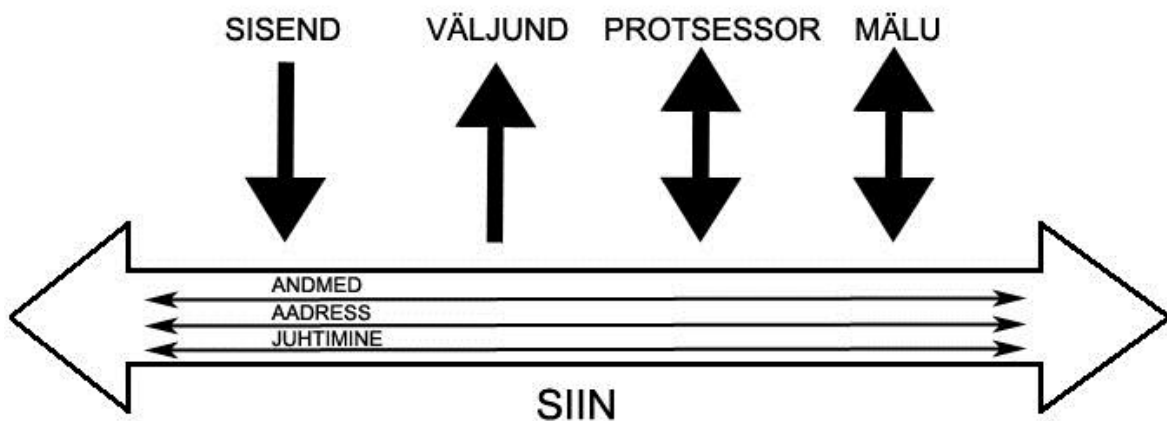
2.2 Reaalaja kella tööpõhimõte

Reaalaja kell RTC-Nx-v2 (mudeli nimetusega DS1307) on väikese energiatarbimisega täielikult kahend-kümnnendkoodis (inglise keeles *binary-coded decimal*) töötav kell-kalender. Kahend-kümnnendkood on arvude salvestamise viis, kus arvu iga kümnnendkoht on eraldi teisendatud kahendkoodi ja salvestatud omaette baidina. Kümnnendkoha salvestamiseks kulub 4 bitti, ülejäänud 4 bitti jäetakse tühjaks või kasutatakse märgi (positiivne või negatiivne) või teise kümnnendkoha hoidmiseks. Näiteks arv 123 esitub kahend-kümnnendkoodis kujul 00000001 00000010 00000011, kus 00000001 vastab kümnnendarvule 1, 00000010 arvule 2 ja 00000011 arvule 3 [32, 35, 36].

Kellas olevaid ja kellast tulevaid andmeid edastatakse järjestikülekandega kasutades I²C protokoll. I²C on kahesuunaline kahesoone line siin (inglise keeles *bidirectional bus*), mille

töötas välja Philips. Siine on vaja selleks, et liigutada infot ühelt seadmelt teisele. Sellist (I^2C) ühendust kasutatakse aeglaste perifeeria-seadmete (väliste seadmete) ühendamiseks näiteks emaplaatide, või telefonidega. I^2C siinis liiguvad andmed mööda traati üksteise järel bittide kaupa mõlemasuunaliselt - s.t andmeid saab lugeda ja kirjutada [32, 37, 38]. Bitt on väikseim informatsiooniühik mille väärtuseks saab olla 0 või 1. Bittide abil saab esitada suvalisi arve kahendsüsteemis. Bait on arvutites kasutatav infoühik, mis koosneb 8 bitist. Seadet, mis edastab andmed siinile, nimetatakse saatjaks (inglise keeles *transmitter*) ning andmeid vastu võtvat seadet vastuvõtjaks (inglise keeles *receiver*). Andmeid juhtivat seadet nimetatakse ülemaks (inglise keeles *master*) ning tema poolt juhivat seadet alluvaks (inglise keeles *slave*) [32]. Reaalaja kella DS1307 puhul on ülemaks näiteks LEGO roboti juhtaju ning alluvaks kell ise.

I^2C siin töötab kahes režiimis. Standardses režiimis töötab siin 100kHz taktsagedusega ning kiires režiimis 400kHz taktsagedusega. Taktsagedus näitab impulsside arvu sekundis ja määrab protsessori töökiiruse [39]. DS1307 suudab andmeid vahetada ainult standardses režiimis ehk 100kHz taktsagedusega [32]. Järgneval joonisel (Joonis 9) on lühidalt ära toodud siini tööpõhimõte.



Joonis 9. Siini tööpõhimõte.

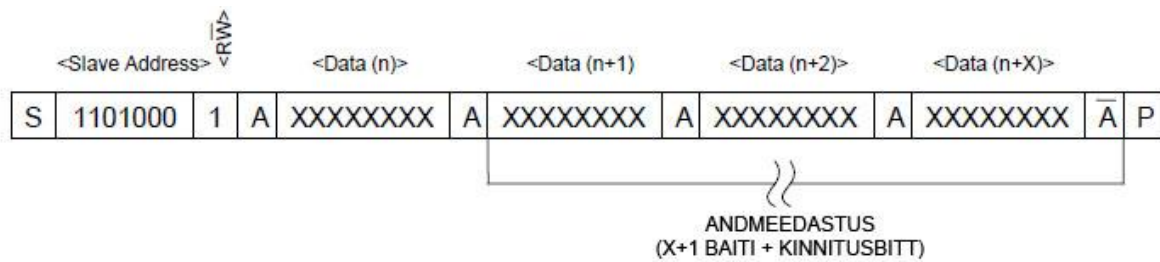
Uut ühendust saab algatada ainult siis, kui siin ei ole hõivatud. Kui kella andmeliin (inglise keeles *clock line*) on andmeedastuse ajal kõrges (HIGH) olekus, peab andmeliin püsima stabiilsena. Andmeliinil toimuvaid muutuseid käsitletakse sellisel juhul kontrollsignaalidena. Kõrge olek tähendab, et kella andmeliini väärtuseks on 1, kui väärtuseks on 0, siis on olek madal (LOW). Andmeliin on stabiilne siis, kui tema väärtus ei

muutu andmeedastuse käigus [32]. Lisas 1 on võimalik lähemalt tutvuda kuidas toimub andmeedastus I²C siinil.

Siini ühendusele on defineeritud järgnevad tingimused [32]:

- **Siini ühendus on vaba** (inglise keeles *Bus not busy*): Andme- ja kella liinid on kõrges (HIGH) olekus.
- **Andmeedastuse alustamine** (inglise keeles *Start data transfer*): Kui andmeliinil toimub muutus kõrgest (HIGH) olekust madalasse (LOW) olekusse ning kella liin on kõrges (HIGH) olekus, siis see tähendab START seisundit.
- **Andmeedastuse lõpetamine** (inglise keeles *Stop data transfer*): Kui andmeliinil toimub muutus madalast (LOW) olekust kõrgesse (HIGH) olekusse ning kella liin on kõrges (HIGH) olekus, siis see tähendab STOP seisundit.
- **Andmete valideerimine** (inglise keeles *Data valid*): Valideerimine on andmete õigsuse ja täielikkuse vastavuse määramise protsess [40]. Andmeplokk on täielik, kui pärast START seisundi tekkimist kella kõrge (HIGH) oleku ajal püsib andmeliin stabiilsena kõrges (HIGH) olekus.
- **Kinnitus** (inglise keeles *Acknowledge*): Vastuvõttev seade peab pärast iga baidi kättesaamist saatma saatjale kinnitusbiti. Ülem genereerib selleks kõrges (HIGH) olekus taktimpulsi, mille ajal peab andmeliin olema stabiilselt madalas (LOW) olekus. Andmeedastuse lõpetamiseks (inglise keeles *end of data*) jätab ülemseade pärast viimase baidi andmete kättesaamist alluvale kinnitusbiti saatmata. Sellisel juhul jätab alluv andmeliini kõrgesse (HIGH) olekusse selleks, et ülem saaks saata lõpetamiskäsu (STOP seisundi käsu).

Seadmetevahelise andmevahetuse loomiseks algatab ülem I²C siinis START seisundi ning lõpetamiseks STOP seisundi. Vahetatavate andmete mahu määrab ülemseade ning START ja STOP seisundi vahel saadetavate baitide hulk ei ole piiratud. Informatsiooni saadetakse mööda siini baidi kaupa (Joonis 10). Iga vastu võetud baidi järel saadab vastuvõtja kinnitusbiti, et on info kätte saanud.



Joonis 10. Andmete lugemine reaalaraja kellast DS1307 [32].

Joonise 10 selgitus:

- S – START seisund;
- A – kinnitusbitt;
- P – STOP seisund;
- \bar{A} – andmete lugemisel pärast viimase baidi kättesaamist kinnitusbitti ei saadeta;
- R/W – andmeid kas loetakse või saadetakse. Lugemise puhul on R/W biti väärtuseks 1.

Siinil on võimalik kahte tüüpi andmeedastus:

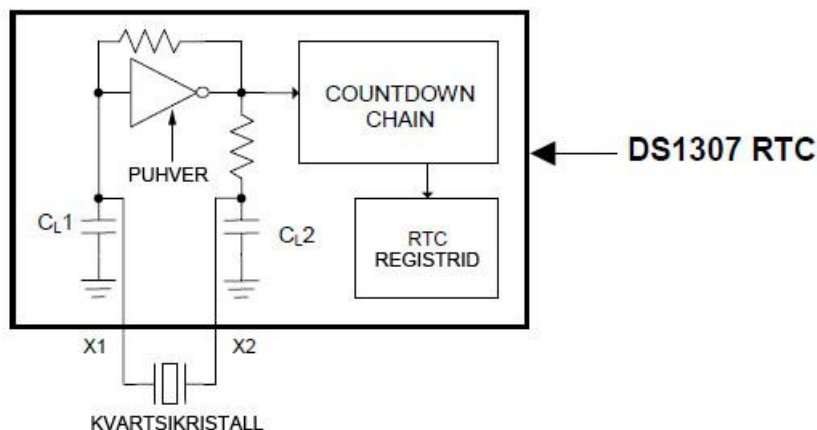
1. **Andmeid saadetakse ülemalt (saatja) alluvale (vastuvõtja):** Esimene saadetud bait on alluva aadress, seejärel järgnevad baidid andmetega. Alluv saadab pärast iga baidi kättesaamist kinnitusbiti. Andmeid edastatakse vastavalt suurima kaaluga bitile (inglise keeles *most significant bit, MSB*), st andmete saatmist alustatakse kõige vasakpoolsemast bitist [41]. Näiteks kui saadetakse binaararvu 11010010, siis on MSB 11010010, järgmisel sammul saadetakse temast parempoolne bitt 11010010 jne.
2. **Andmeid saadetakse alluvalt (saatja) ülemale (vastuvõtja):** Esimese baidi saadab ülem ning selleks on alluva aadress. Seejärel edastab alluv kinnitusbiti ning hakkab saatma andmeid baitide kaupa. Pärast iga baidi kättesaamist, vastab ülem kinnitusbitiga. Pärast viimase baidi kättesaamist kinnitusbitti enam ei saadeta.

Ülemseade genereerib kõik taktimpulsid ning START ja STOP seisundid. Andmete ülekanne lõpeb STOP seisundiga või olukorras kus edastatakse mitu START seisundit järjest. Kuna START seisundiga peaks algama uute andmete edastus, aga eelnevalt pole STOP seisundit antud, siis siini ei avata.

Alluva aadress on esimene saadetav bait. See koosneb 7 bitisest DS1307 aadressist milleks on 1101000. Viimane, 8. bitt, määrab andmeedastussuuna (lugemine/kirjutamine). Lugemise puhul on selleks 1 ja kirjutamise puhul 0. Seega, kui hakatakse reaallaja kellast andmeid lugema, on ülemalt alluvale saadetavaks aadressiks 1101000**1**. Joonisel 10 on näha andmete lugemine reaallaja kellast.

Reaallaja kella DS1307 on sisse ehitatud vooluandur, mis lülitab seadme voolukatkestuse tekkimisel automaatselt patareitoitele. Kell tarbib patarei säästurežiimis töötades vähem kui 0,0000005 A (500 nA nano-Amprit). Kui kell saab voolu väliselt toiteallikalt ning peaks juhtuma, et pinge langeb allapoole $1,25 \cdot$ patareitoite nimipinge, siis katkestatakse andmeedastus. Eelnevalt vastuvõetud andmed tühistatakse vältimaks valede andmete kirjutamist seadme mälusse. Kui pinge langeb allapoole patareitoite nimipinget, siis lülitub seade ümber patarei säästurežiimile. Tagasi välisele toitele minnakse siis, kui pinge on suurem kui patareitoite nimipinge + 0,2V ning andmeid hakatakse uuesti edastama olukorras, kus pinge on kõrgem kui $1,25 \cdot$ patareitoite nimipinge.

Reaallaja kell DS1307 kasutab aja mõõtmiseks kristall-ostsillaatorit (Joonis 11). Ostsillaator on ühtlaselt võnkuv generaator. Ühtlase võnke saavutamiseks kasutatakse kristalli, milleks on peamiselt kvartsikristall [42]. DS1307 kasutab välist kvartsikristall mille võnkesagedus on 32,768kHz. Ostsillaatori vooluring ei vaja töötamiseks väliseid takisteid ega kondensaatoreid. Kella korrektsus sõltub kristalli võnkumise sagedusest ning ostsillaatori ja kristalli vahelise koormusmahtuvuse sobivusest. Mõõtevigu tekitavad kristalli võnkesageduse muutused, mis on tingitud temperatuuri kõikumisest ja välisest mürast (elektrisignaali kõikumine).



Joonis 11. Reaallaja kella DS1307 ostsillaatori loogikaskeem [32].

Joonisel 11 olevate mõistete kirjeldus:

- X1, X2 - kvartsikristalli ühendused ostsillaatoriga;
- C_{L1} , C_{L2} - kvartsikristalli elektrimahtuvus on 12,5 pF (pico (10^{-12}) Faradit). Elektrimahtuvus on füüsikaline suurus, mis iseloomustab keha võimet säilitada elektrilaengut [43].

Reaalaja kellal on 56 baiti mälu. DS1307 andmeid hoitakse kella- ja mäluregistris (Lisa 2). Kella registre aadressid on 00H kuni 07H, mäluregistrid on aadressidel 08H kuni 3FH. Kella aeg ja kuupäev väärtustatakse vastavatele registribaitidele kirjutamisel. Väärtused on kahend-kümnenndkoodis. Registri 00H 7. bitt on kella seisma paneku bit (inglise keeles *clock halt*, CH). Kui selle biti väärtuseks on 1, siis on ostsillaator välja lülitatud ning kell ei mõõda aega.

Reaalaja kell töötab nii 12 kui 24 tunni formaadis. Tunni registri (02H) 6. bitt võimaldab valida formaati ning 5. bitt määrab AM/PM tähistuse (AM on 0 ja PM on 1). Kui on valitud 24 tunni formaat, siis määrab 5. bitt tunnid vahemikus 20 - 23.

Andmete lugemisel või kirjutamisel kasutatakse kella sisemiste registre uuendamise ajal vigaste sisestuste vältimiseks teisejärgulisi puhvrid. Puhver on mäluosa ajutiste andmete salvestamiseks [44]. Puhvrid sünkroniseeritakse kellaga kellaaja ja kuupäeva registre lugemisel. Kuna kell mõõdab pidevalt aega, siis toimuvad registris muudatused. Registritest pidevalt info lugemine oleks ajakulukas. Puhvrite kasutamine võimaldab lugeda kella informatsiooni registre asemel otse puhvritest ning kella tööd ei häirita pidevate infopäringutega

Register 07H (Joonis 12) on kella kontrollregister.

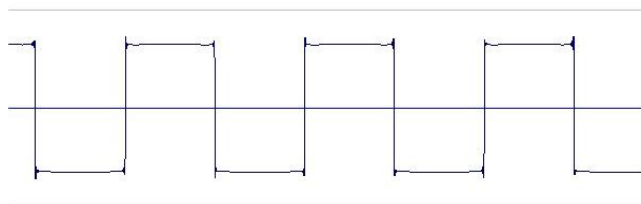
BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
OUT	0	0	SQWE	0	0	RS1	RS0

Joonis 12. Reaalaja kella DS1307 kontrollregister.

Kontrollregistri selgitus:

- **Bitt 7** kontrollib väljundit. Kui 7. biti väärtus on 0, siis midagi ei väljastata.
- **Bitt 4** kontrollib ruutlaine sagedust. Kui 4. biti väärtus on 1, siis väljastatakse ostsillaatori väljund. Ostsillaator tekitab ruutlaine (inglise keeles *square wave*)

sagedusega 1Hz, 4kHz, 8kHz või 32kHz. Ruutlaine taktsagedus sõltub 1. ja 2. biti väärtustest. Ruutlaine (Joonis 13) on enamasti elektroonikas ning signaalitöötluses kasutatav laine.



Joonis 13. Ruutlaine [45].

- **Bitid 0 ja 1** kontrollivad ruutlaine sageduse kiirust, kui 4. biti väärtuseks on 1. Järgnevalt tabel 1 taktsagedusest vastavalt 0. ja 1. biti väärtustele.

Tabel 1. Reaalaja kella DS1307 ruutlaine taktsagedus vastavalt kontrollregistri 1. ja 0. biti väärtusele [32].

Bitt 1	Bitt 0	Ruutlaine taktsagedus
0	0	1Hz
0	1	4,096kHz
1	0	8,192kHz
1	1	32,768kHz

RTC-Nx-v2 ei vaja eraldi kalibreerimist. Kellaaja lähtestamiseks tuleb kirjutada õigetesse registritesse või ühendada andur arvutiga ning kasutada NXT-G keskkonnas reaalaja kella plokki.

Selles punktis anti täpsem ülevaade reaalaja kella töötamise põhimõttest. Järgnev punkt kirjeldab kasutajale reaalaja kella DS1307 maksimumnäitajaid.

2.3 Oluline jälgida

Reaalaja kella kasutamisel tuleb jälgida selle anduri maksimumnäitajaid, kus vahemikus see veel töötab. DS1307 suudab töötada patareitoitel temperatuuril 25°C üle 10 aasta.

Reaalaja kella maksimumnäitajad:

- Pingevahemik, et kontaktid töötaksid: -0,5V kuni +7,0V;
- Töötemperatuur (mittekondenseerivas keskkonnas):
 - Tavakasutusse mõeldud kell: 0°C kuni +70°C;
 - Tööstuslik kell: -40°C kuni +85°C;
- Ladustamistemperatuur: -55°C kuni +125°C;
- Jootmistemperatuur: +260°C juures kuni 10 sekundit.

Maksimumnäitajate ületamine võib põhjustada seadmele jäädavaid kahjustusi. Kui kasutada seadet pikema aja jooksul tema võimete piiril, võib tekkida probleeme anduri täpsusega.

Reaalaja kellaga on võimalik lahendada erinevaid ülesandeid. Järgnevalt kirjeldatakse DS1307 kella programmeerimisvõimalusi.

2.4 Reaalaja kella programmeerimine

Käesolevas töös käsitletavat reaalajalist kella on võimalik programmeerida kasutades kolme erinevat programmeerimiskeelt:

- NXT-G – graafiline programmeerimiskeskond LEGO MINDSTORMS robotite tarbeks (LEGO MINDSTORMS Education NXT).
- RobotC - Carnegie Mellon Robotics Academy poolt välja töötatud C keelel põhinev programmeerimiskeel.
- NBC - *Next Byte Codes* (NBC) on lihtne avatud lähtekoodiga programmeerimiskeel NXT anduri programmeerimiseks [46].

Käesolevas bakalaureusetöös keskendutakse NXT-G keskkonnale, kuna seda kasutatakse kooliroboti kursuste raames kõige enam. See keskkond on algajale robotikahuvilisele kõige lihtsamini omandatav kuna ei eelda eelnevate programmeerimiskogemuste olemasolu.

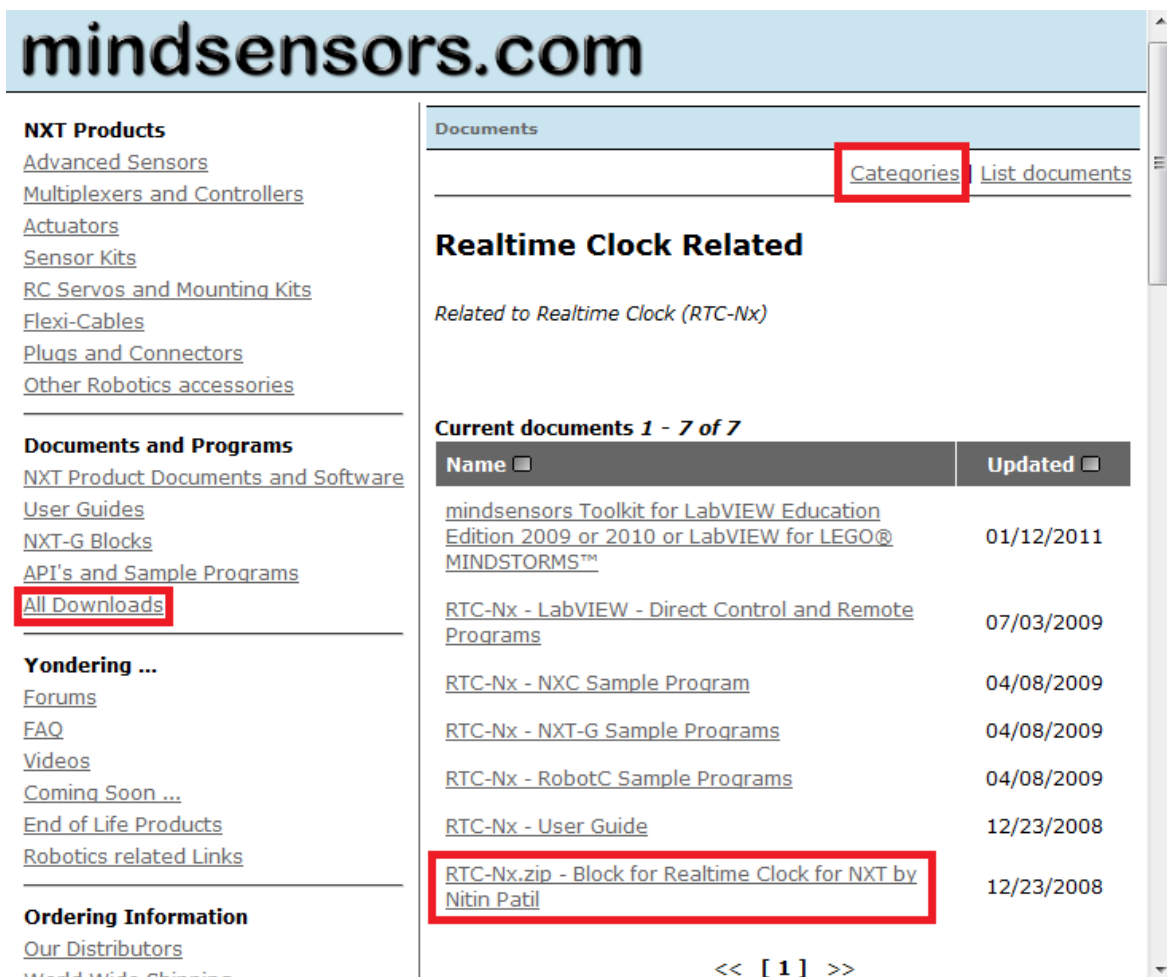
Järgnevalt esitatakse ülevaade reaalajalise kella programmeerimisploki lisamisest NXT-G keskkonda.

2.4.1 Reaalaja kella programmeerimisploki lisamine LEGO MINDSTORMS Education NXT keskkonda

Eelnevalt peab olema arvutisse paigaldatud LEGO MINDSTORMS Education NXT v2.0. Kuna algselt puudub programmis reaalajalise kella programmeerimiseks vajalik plokk, tuleb see käsitsi lisada. Nimetatud plokki on võimalik alla laadida firma Mindsensors kodulehelt [47]. Mindsensors tegeleb isetehtud sensorite disainimise ja valmistamisega robotihuvilistele.

Ploki allalaadimiseks tuleb käituda järgnevalt:

- Internetilehekülje [47] menüüst valida *All Downloads* ning seejärel *Categories*.
- Järgnevalt tuleb üles otsida jaotis *Realtime Clock Related* kust võib leida reaalajalise kella *RTC-Nx*-ga seonduvad failid.
- Alla tuleb laadida *RTC-Nx.zip* fail (Joonis 14) [48].



mindsensors.com

NXT Products
[Advanced Sensors](#)
[Multiplexers and Controllers](#)
[Actuators](#)
[Sensor Kits](#)
[RC Servos and Mounting Kits](#)
[Flexi-Cables](#)
[Plugs and Connectors](#)
[Other Robotics accessories](#)

Documents and Programs
[NXT Product Documents and Software](#)
[User Guides](#)
[NXT-G Blocks](#)
[API's and Sample Programs](#)
[All Downloads](#)

Yondering ...
[Forums](#)
[FAQ](#)
[Videos](#)
[Coming Soon ...](#)
[End of Life Products](#)
[Robotics related Links](#)

Ordering Information
[Our Distributors](#)
[World Wide Shipping](#)

Documents

Categories | [List documents](#)

Realtime Clock Related

Related to Realtime Clock (RTC-Nx)

Current documents 1 - 7 of 7

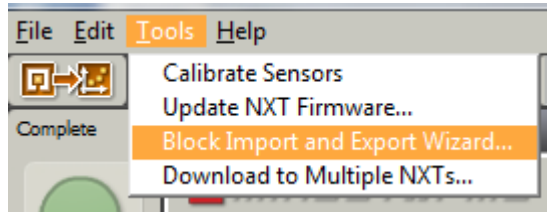
Name <input type="checkbox"/>	Updated <input type="checkbox"/>
mindsensors Toolkit for LabVIEW Education Edition 2009 or 2010 or LabVIEW for LEGO® MINDSTORMS™	01/12/2011
RTC-Nx - LabVIEW - Direct Control and Remote Programs	07/03/2009
RTC-Nx - NXC Sample Program	04/08/2009
RTC-Nx - NXT-G Sample Programs	04/08/2009
RTC-Nx - RobotC Sample Programs	04/08/2009
RTC-Nx - User Guide	12/23/2008
RTC-Nx.zip - Block for Realtime Clock for NXT by Nitin Patil	12/23/2008

<< [1] >>

Joonis 14. Reaalaja kella programmeerimisploki allalaadimine firma Mindsensors koduleheküljelt.

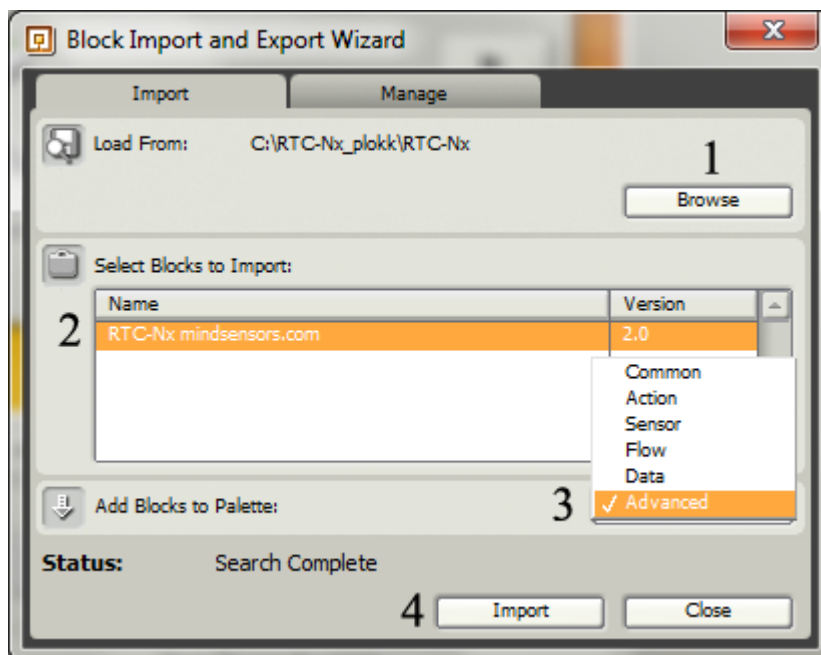
Seejärel lisame plokki NXT-G keskkonda. Selleks tuleb:

- teostada *RTC-Nx.zip* lahtipakkimine suvalisse kausta (nt C:\RTC-Nx_plokk).
- käivitada LEGO MINDSTORMS Education NXT v2.0.
- avanenud aknas valida menüüribalt *Tools* ning seejärel *Block Import and Export Wizard* (Joonis 15).



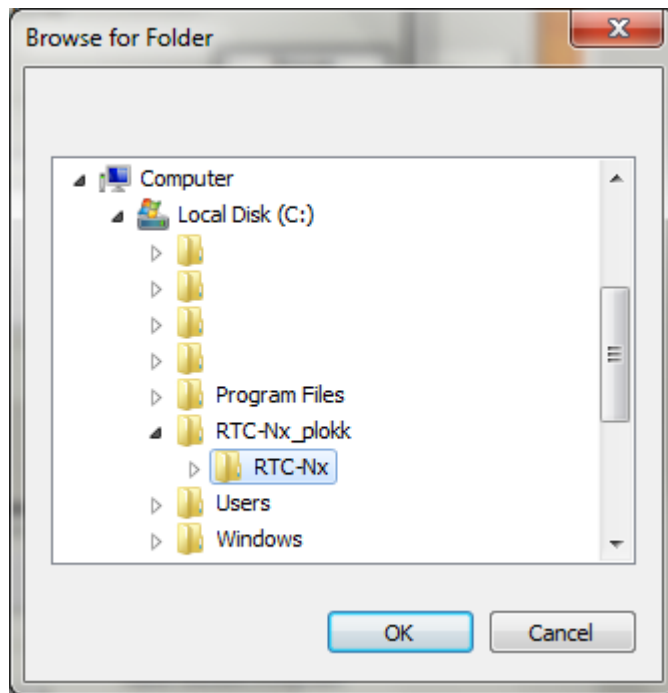
Joonis 15. Reaalaja kella plokki lisamine NXT-G keskkonda.

- Uues aknas avaneb plokkide importimise ja eksportimise viisard (inglise keeles *wizard*) (Joonis 16).



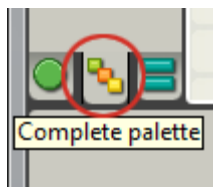
Joonis 16. Plokki lisamise viisard NXT-G keskkonnas.

- Viisardis tuleks käituda järgnevalt:
 1. Esmalt vajutada nupule *Browse*.
 - Avaneb uus aken, kus tuleb valida vastav kataloog, kuhu eelnevalt allalaaditud plokk sai lahti pakitud. Praeguse näite puhul on selleks C:\RTC-Nx_plokk\RTC-Nx (Joonis 17).



Joonis 17. Reaalaja kella ploki kataloogi valimine.

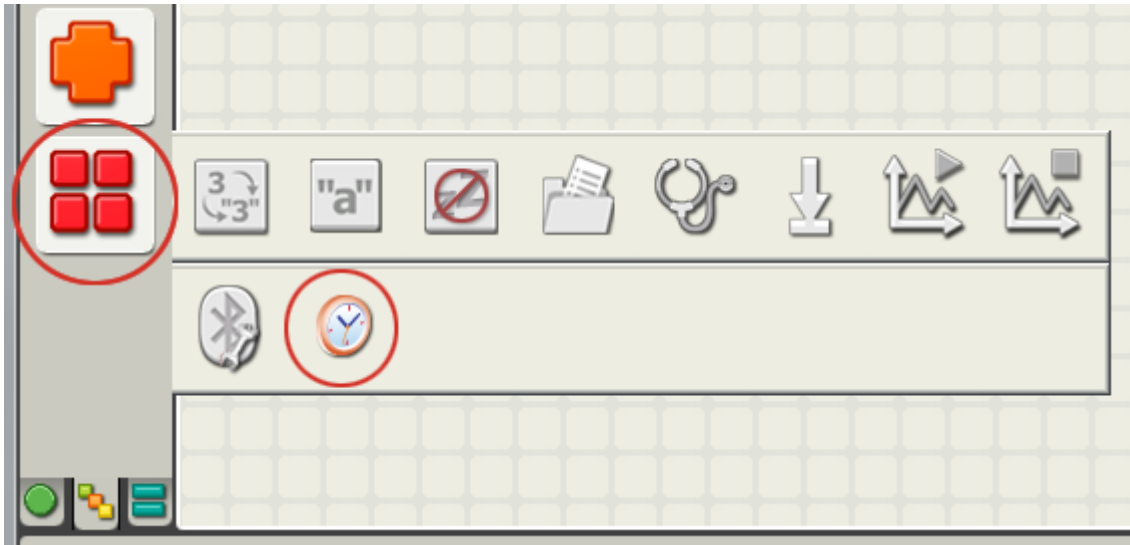
2. Seejärel selekteerida vastav plokk.
3. Soovi korral saab valida, millise alamjaotuse alla lisatav plokk pannakse. Vaikimisi on selleks edasijõudnutele (inglise keeles *Advanced*) mõeldud jaotus täieliku paletti (inglise keeles *Complete palette*) (Joonis 18) all.



Joonis 18. Reaalaja kella plokk on kättesaadav täieliku paletti alt.

4. Ploki lisamiseks vajutada nupule *Import*.

Edaspidi on plokk programmis kättesaadav vastavas kasutaja poolt valitud alamjaotuses. Reaalaja kella plokk on tähistatud kella ikooniga (Joonis 19).



Joonis 19. Reaalaja kella plokk NXT-G keskkonnas.

Plokki on vaja programmi lisada vaid korra. Järgmistel kordadel on see kättesaadav vastava alamjaotuse alt. Järgnevalt ülevaade ploki omadustest NXT-G keskkonnas.

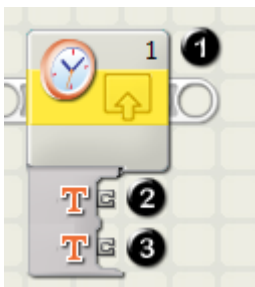
2.4.2 RTC-Nx ploki kuvaseaded

Kui RTC-Nx programmeerimisblokk on NXT-G keskkonda imporditud, siis tekib täieliku paletti (inglise keeles *Complete palette*) alla vastavalt kasutaja poolt valitud alamjaotusse kella ikooniga plokk (Joonis 20).



Joonis 20. Reaalaja kella plokk NXT-G keskkonnas [49].

Kui asetada anduri plokk töölauale, on nähtavad algsed kuvaseaded (inglise keeles *display settings*) (Joonis 21).



Joonis 21. Reaalaja kella algsed kuvaseaded [49].

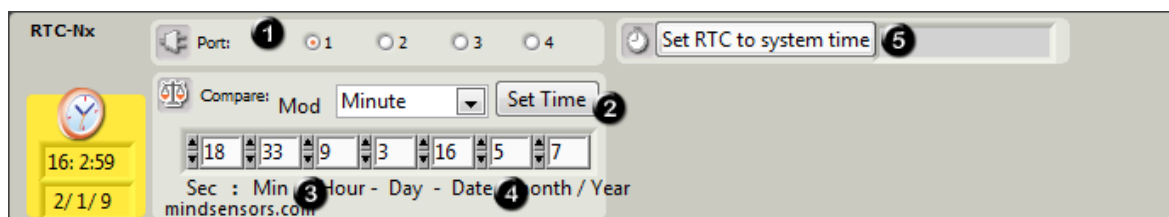
Joonisel 21 olevate komponentide kirjeldus [49]:

1. Paremas ülemises nurgas olev number näitab, millisest NXT pesast andmeid loetakse. Kasutaja saab pesa numbrit juhtpaneelist muuta.
2. Ülemise pistiku väljundiks on aeg sõnena, mis on 24 tunni formaadis (TT:MM:SS). Kui RTC seadmelt andmete lugemisele tekib viga, tagastab plokk aja asemel teate „Error“ või „99“. Sellisel juhul tuleks kontrollida, kas andur on ühendatud õigesse porti.
3. Teine pistik väljastab kuupäeva sõnena (PP-KK-AA).

Järgnevalt kirjeldatakse kella plokki konfigureerimist.

2.4.3 RTC-Nx plokki konfigureerimine

Anduri võimalustega lähemalt tutvumiseks tuleb üle vaadata selle juhtpaneel. Valides töölaual reaalaaja kella plokk, tekib ekraani alaossa anduri juhtpaneel (Joonis 22).



Joonis 22. Reaalaaja kella juhtpaneel LEGO MINDSTORMS Education NXT v2.0 programmeerimiskeskonnas [49].

Komponentide kirjeldus:

1. Pesa (*Port*) – võimaldab valida, millisesse pesasse RTC-Nx on ühendatud.
2. Määra aeg (*Set Time*) nupu abil on võimalik määrata võrdlusmassiivi (*compare array*) andmeteks arvutis olev hetke kellaaeg ja kuupäev.
3. Kellaaja võrdlemiseks on vaja määrata sekundid, minutid ja tunnid.
4. Kuupäeva võrdlemiseks on vaja määrata kuupäev, kuu ja aasta (kuna aastat saab määrata ainult 21. sajandi sees, siis kirjutatakse aastaarv 2-kohalise arvuna, nt aasta 2067 märgitakse arvuna 67).
 - Selleks, et võrrelda sekundeid, minuteid, tunde, kuupäeva, kuud, aastat, peab *Mod* väljal valima vastava väärtuse (*Seconds*, et võrrelda sekundeid, *Minute*, et võrrelda minuteid, jne).

- Nii pannakse tööle nn taimer, mis hakkab tööle vastaval kasutaja poolt valitud ajahetkel. Selle tulemusena on loogikatehte (Yes/No) pistikus väljundiks tõene väärtus vastavalt valitud võrdlusühikule. Näiteks kui võrdluseks kasutati väärtust *Second*, on pistikus ühe sekundi jooksul väärtus tõene, kui võrdluseks kasutada väärtust *Minute*, on pistiku väärtus tõene ühe minuti jne.
5. Määra reaalaja kellale süsteemi aeg (*Set RTC to system time*) võimaldab RTC sensorit arvuti ajaga sünkroniseerida (24 tunni formaadis).

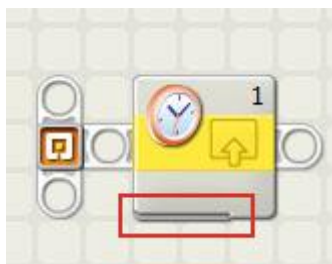
Enne sünkroniseerimist tuleks silmas pidada järgnevat:

- jälgida, et RTC-Nx sensor oleks ühendatud samasse pesasse nagu on valitud plokis;
- kontrollida, et NXT juhtplokk oleks arvutiga korrektselt ühendatud (läbi USB või sinihamba);
- kasutaja peab kontrollima, et NXT ei käitleks ühtegi programmi;
- pärast sünkroniseerimise nupule vajutamist kulub värskendamiseks umbes 10 sekundit.

Järgnevalt tutvustatakse reaalaja kella ploki infotulpa.

2.4.4 RTC-Nx ploki infotulp

Klikkides reaalaja kella ploki allaserva (Joonis 23), ilmuvad nähtavale kõik tema infot hoidvad pistikud (Joonis 24).



Joonis 23. Riba, kuhu vajutades avanevad reaalaja kella kõik pistikud.



Joonis 24. Reaalaja kella plokk avatud infotulbaga.

Joonise 24 selgitus:

1. Näitab numbrit millisesse NXT pesasse on reaalaja kell ühendatud;
2. Režiimi valimine;
3. Tõene/väär - võrdlusfunktsiooni kontroll;
4. Kellaaeg formaadis TT:MM:SS;
5. Kuupäev formaadis PP-KK-AA;
6. Sekundivälja väärtus;
7. Minutivälja väärtus;
8. Tunnivälja väärtus;
9. Päevavälja väärtus;
- a) Kuupäevavälja väärtus;
- b) Kuuvälja väärtus;
- c) Aastavälja väärtus;
- d) Ajamassiivi võrdlus (NXT-G ei toeta);

Selleks, et joonisest 24 paremat ülevaadet saada, selgitatakse vastavate pistikute omadusi lähemalt Lisas 3.

Käesolevas peatükis selgitati lähemalt reaalaaja kella RTC-Nx-v2 tööpõhimõtet ning kirjeldati anduri programmeerimist LEGO MINDSTORMS Education NXT v2.0 programmeerimiskeskkonnas. Järgmine peatükk keskendub andurile mõeldud ülesannetele.

3. Ülesanded reaalaraja kella RTC-Nx jaoks

Eelnevates peatükkides tutvustati lugejale kella ajalugu ning reaalaraja kella omadusi ja tööpõhimõtet. Käesolev peatükk on mõeldud reaalaraja kella abil praktiliste ülesannete lahendamisele. Ülesanded on erineva raskusastmega ning neile kõigile on lisatud võimalik lahenduskäik. Ülesanded on loodud kasutades LEGO MINDSTORMS Education NXT v2.0 programmeerimiskeskonda.

Kõik ülesanded on sarnase struktuuriga, mis peaks hõlbustama nende hilisemat kasutamist õppematerjalina. Ülesanded koosnevad järgmistest osadest:

- Tase - näitab millise tasemega õpilastele võiks ülesanne sobilik olla.
 - Eesmärk - kogemused, teadmised mida ülesande läbija soovitatavalt omandab.
 - Ülesande täitmiseks vajalik - vajalikud vahendid ülesande lahendamiseks.
 - Ülesande püstitus - ülesande teksti võimalikult huvitav esitus. Kirjeldab probleemi olemust millele kasutaja hakkab lahendust leidma.
 - Lahenduse idee - annab esialgse idee kuidas ülesande lahendamist alustada.
 - Üks võimalik lahendusvariant - üksikasjalik selgitus ülesande lahendusest.
- Ülesanded on loodud NXT-G programmeerimiskeskonnas ning lahenduse failid asuvad bakalaureusetööga kaasas oleval CD plaadil [Lisa 4].
- Tekkida võivad probleemid ning nende lahendamine - lahendamise käigus tekkivate probleemide lahendused. Mõnel juhul ka ülesandega seonduv lisainfo .

Mõne ülesande puhul võib ülesande lõpust leida selle edasiarenduse. See on seotud algse ülesandega, kuid on sellest keerukam. Tärniga ülesanne on mõeldud pisut usinamatele õpilastele, et pakkuda neile lisategevust.

3.1 Ülesanne 1 - Anduri kasutamine kellana

Tase

Lihtne, sobib algajale reaalaraja kellaga tutvumiseks.

Eesmärk

Ülesande eesmärgiks on näidata kasutajale, et reaalaraja kell töötab. Kellaaeg väljastatakse reaalarajas ekraanile.

Ülesande täitmiseks vajalik:

- LEGO MINDSTORMS NXT juhtplokk
- RTC-Nx reaalaaja kell
- Ühendusjuhtmed
- Vajadusel USB juhe arvuti ning juhtploki ühendamiseks (saab kasutada ka sinihammast)
- LEGO klotsid
- Programmeerimiskeskond NXT-G (LEGO MINDSTORMS Education NXT)
- RTC-Nx plokk NXT-G keskkonnas
- Puuteandur

Ülesande püstitus

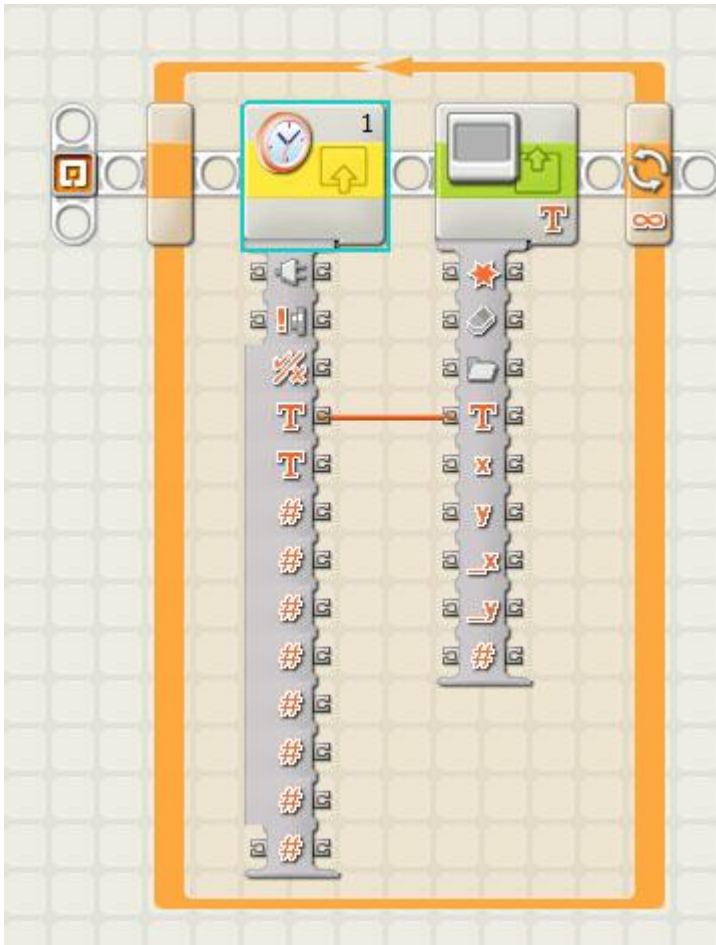
Ants unustas käekella koju, aga tal on vaja hädasti õiget aega teada, et vanaemale bussijaama vastu minna. Õnneks on tal kaasas NXT roboti juhtplokk ning reaalaajaline kell. Reaalaajaline kell kahjuks ise aega ei näita. Aita Antsul käepäraste vahendite abil omaile kell teha.

Lahenduse idee

Kasutada tuleks reaalaaja kella programmeerimise plokki ning ekraanile kuvamise plokki. Programm peab töötama tsüklis, et kellaaega pidevalt uuendataks.

Üks võimalik lahendusvariant

Enne lahenduse kallala asumist tuleks ühendada reaalaaja kell NXT juhtaju esimesse pesasse ning juhtaju ühendada arvutiga. Esimese ülesande lahendus on nähtav joonisel 25. Kellaaja kuvamiseks ekraanile on tarvis ühendada reaalaaja kella ploki ajaväljund ekraani tekstiväljundiga. Ajaväljund on sõne tüüpi ning ekraanile väljastamiseks ei pea seda ümber teisendama. Programm peab töötamiseks olema kindlasti tsüklis.



Joonis 25. Ülesande nr 1 üks võimalik lahendusvariant.

Lahendusfail: ylesanne1.rbt [Lisa 4].

Tekkida võivad probleemid ning nende lahendamine

Jälgima peaks, et reaalaaja kell oleks ühendatud õigesse NXT juhtaju pesasse. Reaalaaja kella võiks sünkroniseerida arvuti kellaajaga.

Ideed ülesande muutmiseks

Lisada programmile võimalus vaadata nii kellaiega kui ka kuupäeva. Lahendamiseks kasutada puuteandurit. Kui vajutatakse puuteandurit, siis näidatakse ekraanil kuupäeva, muul ajal endiselt reaalaajas kellaiega.

Lahendusfail: ylesanne1_1.rbt [Lisa 4].

3.2 Ülesanne 2 - Andmete kogumine

Tase

Keskmine, sobib anduriga lähemalt tutvumiseks.

Eesmärk

Tutvustada kasutajale võimalust reaalaaja kellaga andmeid koguda. Kasutaja saab ülevaate erinevate andurite koos töötamise võimalustest.

Ülesande täitmiseks vajalik

- LEGO MINDSTORMS NXT juhtplokk
- RTC-Nx reaalaaja kell
- Ühendusjuhtmed
- Vajadusel USB juhe arvuti ning juhtploki ühendamiseks (saab kasutada ka sinihammast)
- LEGO klotsid
- Programmeerimiskeskond NXT-G (LEGO MINDSTORMS Education NXT)
- RTC-Nx plokk NXT-G keskkonnas
- Heliandur

Ülesande püstitus

Ants väidab, et tema vend Margus norskab öösel. Margus ei usu sellist laimu. Ants otsustab ehitada roboti, mis salvestaks nendel ajahetkedel, kui Margus norskab, kellaaja.

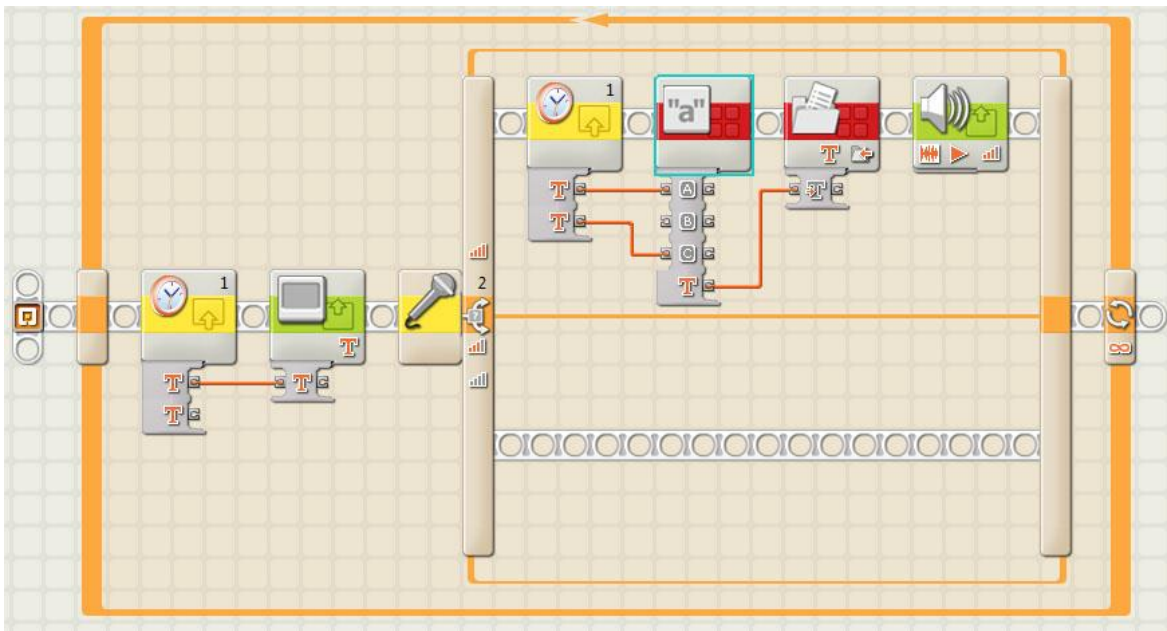
Lahenduse idee

Tuleks kasutada reaalaaja kella koos helianduriga ning salvestada andmed faili. Heliandurit ei tohiks seada liialt tundlikuks. Lisaks kellaajale võiks salvestada ka kuupäeva.

Üks võimalik lahendusvariant

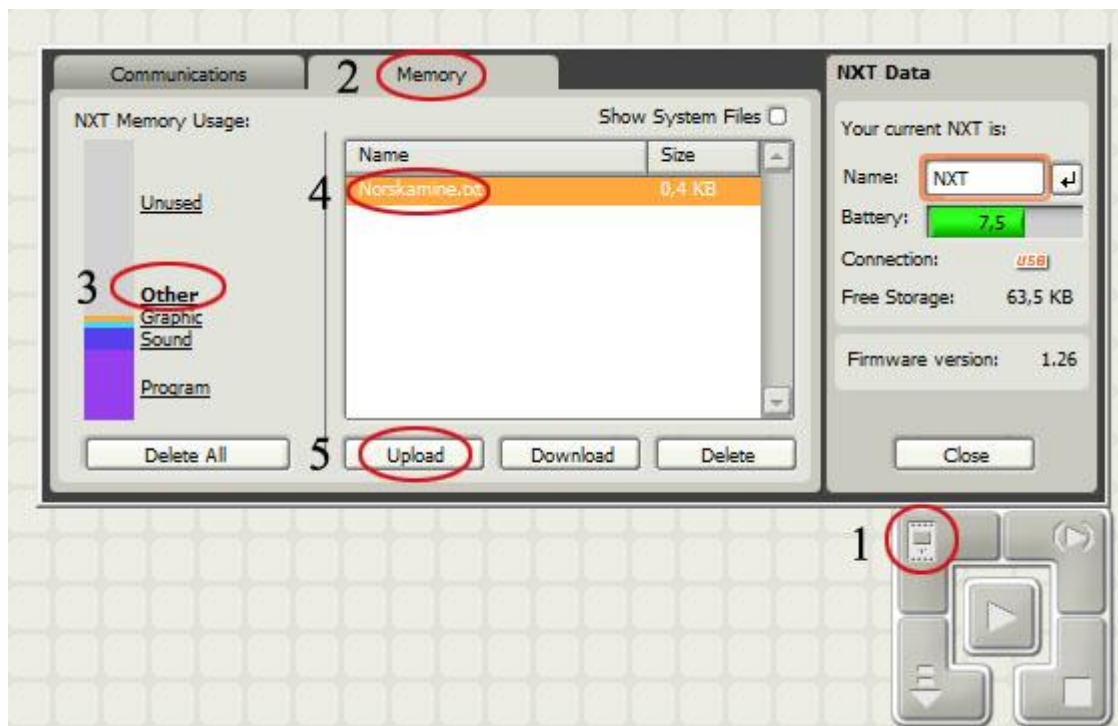
Lahendus on nähtav joonisel 26. Kellaaja salvestamiseks tuleb kasutada faili (inglise keeles *File Access*) plokki. Lisaks sellele peab programmis olema tõeväärtust kontrolliv plokk, mis tõese väärtuse korral andmed faili salvestab. Kindlasti peaks programm olema tsüklis. Ülesandele on lisatud mugavuse mõttes reaalaajas uuenev kellaaja näit. Kasutatud on lüliti (inglise keeles *Switch*), mis jälgib helianduri näitu. Iga kord kui heliandurile määratud

helitugevus ületatakse muutub lülitis väärtus tõseks ning täidetakse ettemääratus ülesandeid. Reaalaja kella kuupäeva ja kellaaja sõned on kokku pandud ühtseks tekstiks kasutades NXT-G tekstiplokki ("a"). Tekstiploki pistik B on jäetud tühjaks, et oleks võimalik panna kellaaja ja kuupäeva sõne vahele, mis dokumenti kirjutatakse, tühik. Heli mängimise plokk on ülesandele lisatud sellepärast, et kasutajal oleks lihtsam aru saada olukorrast millal detsipellide tase ruumis ületab heliandurile määratud taseme. Seega saab kasutaja helilist tagasisidet, kui infot salvestatakse. Soovitav on salvestatavale failile määrata eriline nimi, et need omavahel segamini ei läheks. Faili nime saab määrata faili salvestamise ploki juhtpaneelilt. Salvestatud faili kättesaamist NXT robotist selgitab joonis 27.



Joonis 26. Ülesande nr 2 üks võimalik lahendusvariant.

Lahendusfail: ylesanne2.rbt [Lisa 4].



Joonis 27. Faili allalaadimine NXT juhtajust arvutisse.

1. Esimese sammuna tuleb vajutada NXT-G programmeerimisekeskkonna paremas alumises nurgas asuvalle *NXT Window* ikoonile.
2. Avaneb joonisega 27 sarnane aken. Selle ülemistest sakkidest tuleb valida väli *Memory*.
3. Seejärel tuleb vasakpoolsest tulbast valida kategooria *Other*.
4. Selle kategooria alt ilmub nähtavale ülesande käigus salvestatud fail mille nimeks praeguse näite puhul sai pandud *Norskamine.txt*.
5. Faili allalaadimiseks arvutisse tuleb vajutada nupule *Upload* ning valida avanevas aknas sobiv salvestuskoht.

Tekkida võivad probleemid ning nende lahendamine

Salvestatud andmete leidmine võib esmakordsel kasutamisel tekitada probleeme. Jälgima peaks, et andurid oleks ühendatud õigetes pesadesse. Tulemused võivad erineda vastavalt heliandurile seatud tundlikkusele.

3.3 Ülesanne 3 - Äratuskell

Tase

Keskmine, sobib anduriga lähemalt tutvumiseks.

Eesmärk

Kasutaja saab ülevaate, kuidas reaalaaja kell ajamassiive võrdleb.

Ülesande täitmiseks vajalik

- LEGO MINDSTORMS NXT juhtplokk
- RTC-Nx reaalaaja kell
- NXT mootorid
- Ühendusjuhtmed
- Vajadusel USB juhe arvuti ning juhtploki ühendamiseks (saab kasutada ka sinihammast)
- LEGO klotsid
- Programmeerimiskeskond NXT-G (LEGO MINDSTORMS Education NXT)
- RTC-Nx plokk NXT-G keskkonnas
- Kauguse andur
- Lambike

Ülesande püstitus

Indrek magab hommikuti sisse ning hilineb seetõttu tihtilugu esimesesse tundi. Klassijuhataja ähvardas järgmise hilinemise korral kutsuda Indreku vanemad kooli. Indrekule sellised ähvardused ei meeldi ja seetõttu otsustas ta ehitada omaale äratuskella, mis ei lase sisse magada. Indrekule ei piisa tavalisest äratuskellast, sest selle vajutab ta lihtsalt kinni. Aita tal ehitada äratuskell, mis sõidab mööda tuba ringi.

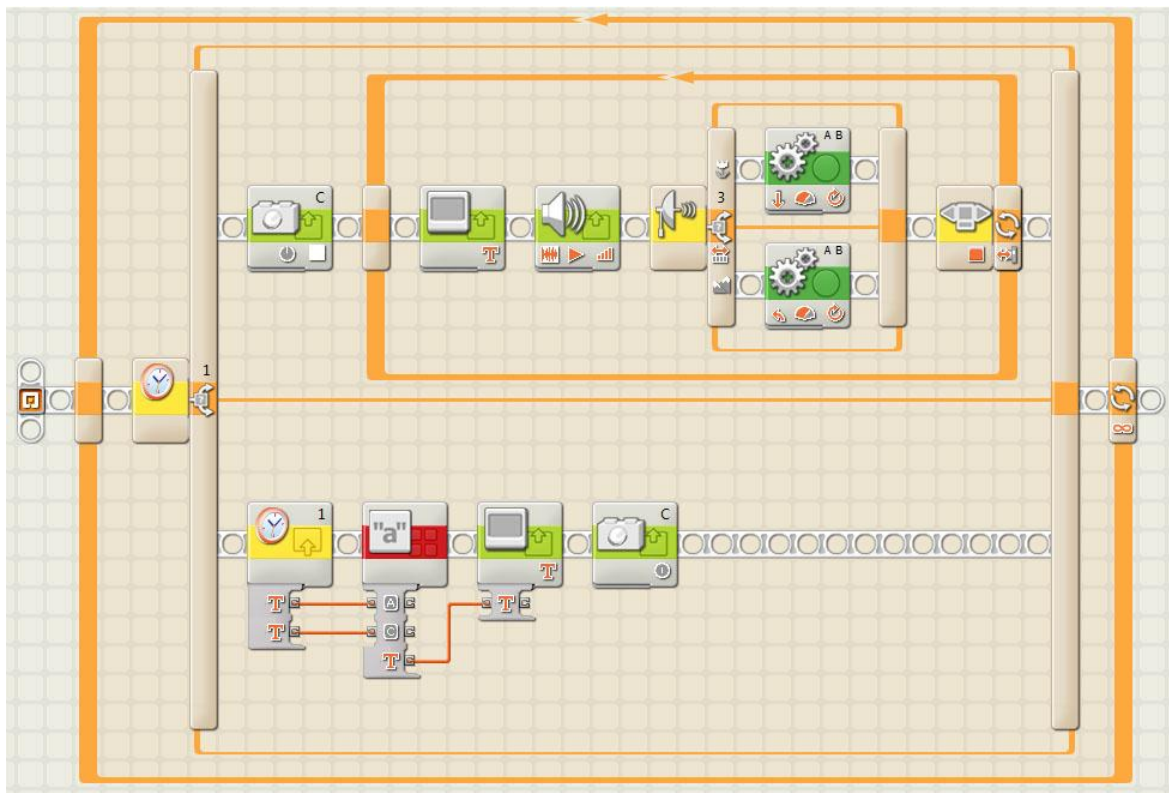
Lahenduse idee

Äratuskell, mis sõidab eest ära. Lisaks sellele võiksid ka tuled põlema minna. Kasutada tuleks reaalaaja kella määramaks õiget äratuse aega ning LEGO MINDSTORMS NXT juhtaju ning mootoreid.

Üks võimalik lahendusvariant

Võimalik lahendusvariant on nähtav Joonisel 28. Kasutatud on lüliti, mis jälgib reaalaaja kella. Reaalaaja kellas tuleb valida võrdlusmassiviks *Time* ja määrata soovitud äratuse aeg vajutades nupule *Set Time*. Ajal, mil võrdlusmassiivi väärtus on väär, näidatakse ekraanil hetke kellaaega ja kuupäeva. Kui kätte jõuab äratuse aeg kuvatakse ekraanile tekst “2RATUS!”, pannakse põlema pesasse C ühendatud lambike, mängitakse heli ning robot

hakkab mööda tuba ringi sõitma. Takistuste vältimiseks kasutatakse kauguse andurit, mis annab robotile käsu tagurdada, kui ollakse mõnele objektile lähemal kui 10 cm. Kogu eelnimetatud tegevus kestab senikaua kuni kasutaja vajutab NXT juhtploki *Enter* nupule.



Joonis 28. Ülesande nr 3 üks võimalik lahendusvariant.

Lahendusfail: ylesanne3.rbt [Lisa 4].

Tekkida võivad probleemid ning nende lahendamine

Peaks jälgima, et robot liiga kaugele ära ei sõidaks. Andurid peavad olema ühendatud õigesti pesadesse. Kellaaja ja kuupäeva kuvamisel tuleks jälgida, et kogu tekst mahuks ekraanile. Kindlasti tuleb jälgida, et kauguse andur oleks sobivale kõrgusele kinnitatud.

3.4 Ülesanne 4 - Eestikeelne väljund

Tase

Keskmine, sobib anduri ja NXT-G programmeerimiskeskonnaga lähemaks tutvumiseks.

Eesmärk

Kasutaja tutvub reaalaaja kella pistikute väärtustega.

Ülesande täitmiseks vajalik

- LEGO MINDSTORMS NXT juhtplokk
- RTC-Nx reaalaja kell
- Ühendusjuhtmed
- Vajadusel USB juhe arvuti ning juhtploki ühendamiseks (saab kasutada ka sinihammast)
- Programmeerimiskeskond NXT-G (LEGO MINDSTORMS Education NXT)
- RTC-Nx plokk NXT-G keskkonnas

Ülesande püstitus

Kersti tahab muuta oma reaalaja kella väljundit NXT juhtaju ekraanil eestikeelseks kuna on numbrilisest väljundist tüdinenud. Alustuseks otsustab ta muuta kuupäeva arvu asemel sõneks. Aita Kerstit oma omandatud teadmistega.

Lahenduse idee

Ekraanile võiks kuvada kuupäeva ning selle järel sõne vastava kuu nimetusega. Selleks on hea kasutada lülitit. Tähele tuleks panna seda, et kasutades inglisekeelset NXT püsivara ei saa täpitahti ekraanile kuvada.

Üks võimalik lahendusvariant

Lahendus võib välja näha umbes nagu joonisel 29. Reaalaja kella kuu pistiku väljund on ühendatud numbrilist väärtust kontrolliva lülitiga. Lülitil tuleb võtta *Flat View* eest linnuke. See võimaldab lisada tingimused, mille tõese väärtuse korral kuvatakse vastav kuu nimetus. Tingimuste lisamiseks tuleb lüliti juhtpaneelil vajutada “+” nupule. Lisada tuleks 12 tingimust kuu nimetuse jaoks ning üks tingimus võiks olla vigase väljundi jaoks. See tähendab, et kui reaalaja kellast lugemisel tekkis tõrge, teavitatakse kasutajat sellest. Iga tingimuse sisse tuleb määrata muutuja (inglise keeles *Variable*) plokk, mis on sõne tüüpi ning kirjutamisrežiimis. Muutujate nimedeks on määratud numbrile vastav kuu nimetus. Kuu nimetuse ekraanile kuvamiseks tuleb lisada väljapoole lülitit uus muutuja, mis on lugemise režiimis ning selle väärtus kuvada ekraanile.

Ülesande püstitus

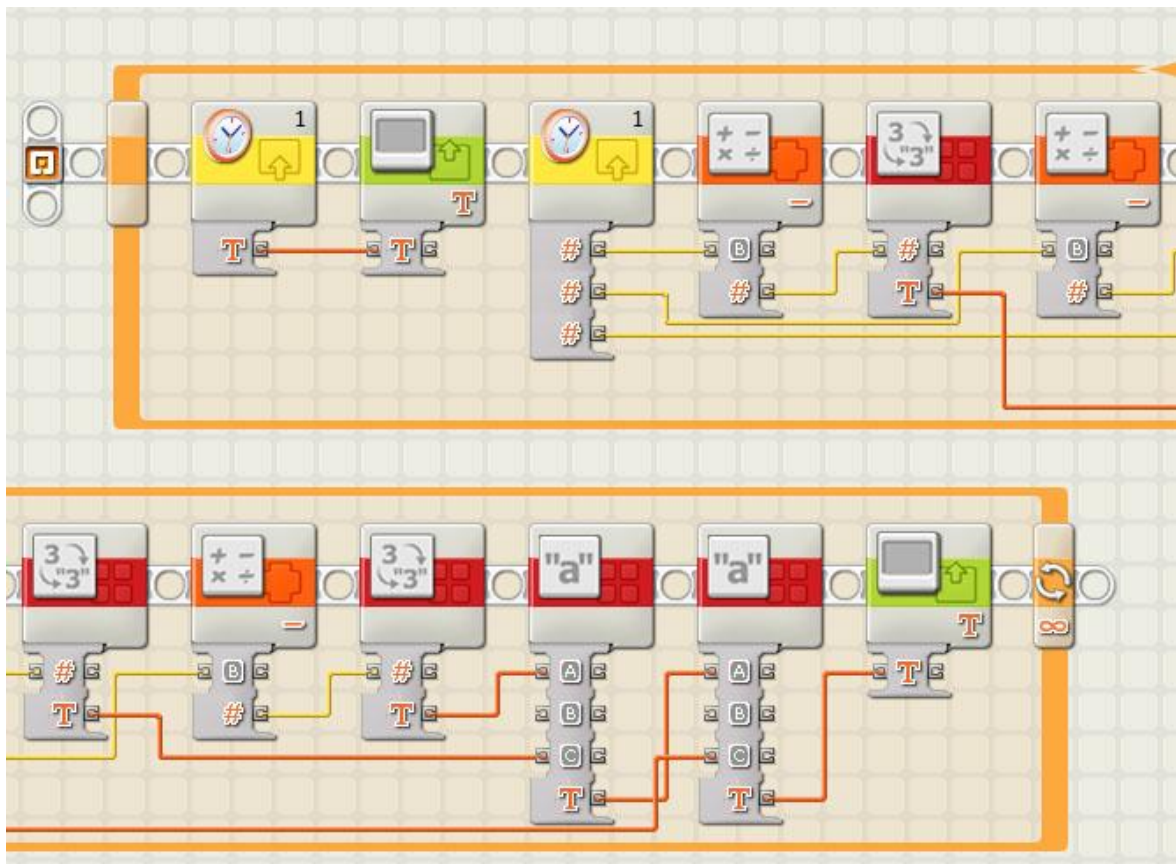
Punamütsike tahab minna külla oma kallile vanaemale, aga väljas on juba pime ja keskööni ei ole palju jäänud. Punamütsike teab, et kuri karu ärkab igal öösel kell 12. Aita Punamütsikesel turvaliselt vanaema majani jõuda.

Lahenduse idee

Reaalaja kella võiks kasutada nagu taimerit, st NXT ekraanile kuvatakse reaalajas allapoole lugedes järelejäänud aeg keskööni.

Üks võimalik lahendusvariant

Võimalik lahendus on nähtav joonisel 30. Ekraanile kuvatakse järelejäänud aeg keskööni ning selle all hetke kellaeg. Ülesande lahendamiseks on kasutatud aritmeetikaplokkide lahutamisfunktsiooni. Numbrilised väljundid teisendatakse sõnedeks ja kuvatakse ekraanile.



Joonis 30. Ülesande nr 5 üks võimalik lahendusvariant.

Lahendusfail: ylesanne5.rbt [Lisa 4].

Tekkida võivad probleemid ning nende lahendamine

Reaalaja kell peab olema ühendatud õigesse pesasse. Lahutamiskutsiooni kasutades tuleks jälgida, et reaalaja kella väärtus on õiges sisendi pistikus. Lahutamiskutsiooni A sisendi väärtused tuleb kasutajal ise määrata (nendeks on 59, 59, 23 vastavalt sekund, minut, tund).

Ideed ülesande muutmiseks

Mõnes teises programmeerimiskeskkonnas on võimalik teha täisväärtuslik taimer.

3.6 Ülesanne 6 - Vilkuvad tuled

Tase

Keskmine

Eesmärk

Kasutaja kasutab reaalaja kella väljundit ülesande lahendamiseks.

Ülesande täitmiseks vajalik

- LEGO MINDSTORMS NXT juhtplokk
- RTC-Nx reaalaja kell
- Ühendusjuhtmed
- Vajadusel USB juhe arvuti ning juhtploki ühendamiseks (saab kasutada ka sinihammast)
- Programmeerimiskeskkond NXT-G (LEGO MINDSTORMS Education NXT)
- RTC-Nx plokk NXT-G keskkonnas
- Lambikesed
- Paaris või paaritu arvu tuvastamise plokk [50].

Ülesande püstitus

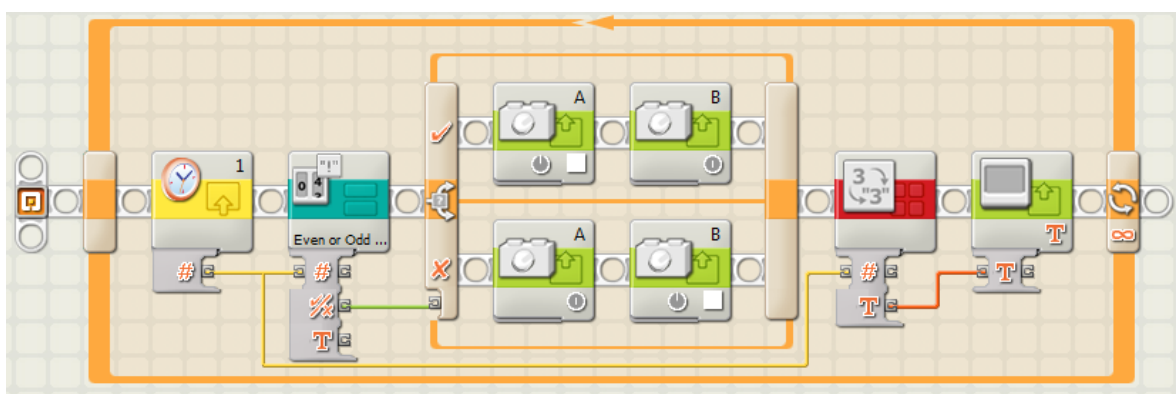
Robin tahab panna oma NXT robotil tuled kella tiksumise taktis vilkuma. Robinil on kõik vajalikud jupid välja otsitud, kuid ta ei oska neid koos tööle panna.

Lahenduse idee

Tuled võiksid minna põlema ja kustuda kordamööda vastavalt sellele, kas reaalaja kella sekundiväljundi väärtuseks on paaris või paaritu arv.

Üks võimalik lahendusvariant

Võimalik lahendus on nähtav joonisel 31. Ülesande lahendamiseks on NXT-G keskkonda lisatud paaris või paaritu arvu tuvastamise plokk [50]. Paarisarvulise väärtuse korral väljastab plokk väärtuse tõene. Tõeväärtus antakse ette lülitile, mis kontrollib tulede põlemist. Kontrolli eesmärgil on ekraanile kuvatud sekundite näit.



Joonis 31. Ülesande nr 6 üks võimalik lahendusvariant.

Lahendusfail: ylesanne6.rbt [Lisa 4].

Tekkida võivad probleemid ning nende lahendamine

Tulukesed ja reaalaja kell peavad olema ühendatud õigetes pesadesse. Tulukeste kustutamiseks peab nad välja lülitama. Vastasel korral nad ei vilgu vaid põlevad ühtlaselt.

3.7 Ülesanne 7 - Seinakell

Tase

Keskmine

Eesmärk

Reaalaja kella kasutamine seinakella juhtimiseks.

Ülesande täitmiseks vajalik

- LEGO MINDSTORMS NXT juhtplokk
- RTC-Nx reaalaaja kell
- NXT mootorid
- Ühendusjuhtmed
- Vajadusel USB juhe arvuti ning juhtploki ühendamiseks (saab kasutada ka sinihammast)
- LEGO klotsid
- Programmeerimiskeskond NXT-G (LEGO MINDSTORMS Education NXT)
- RTC-Nx plokk NXT-G keskkonnas

Ülesande püstitus

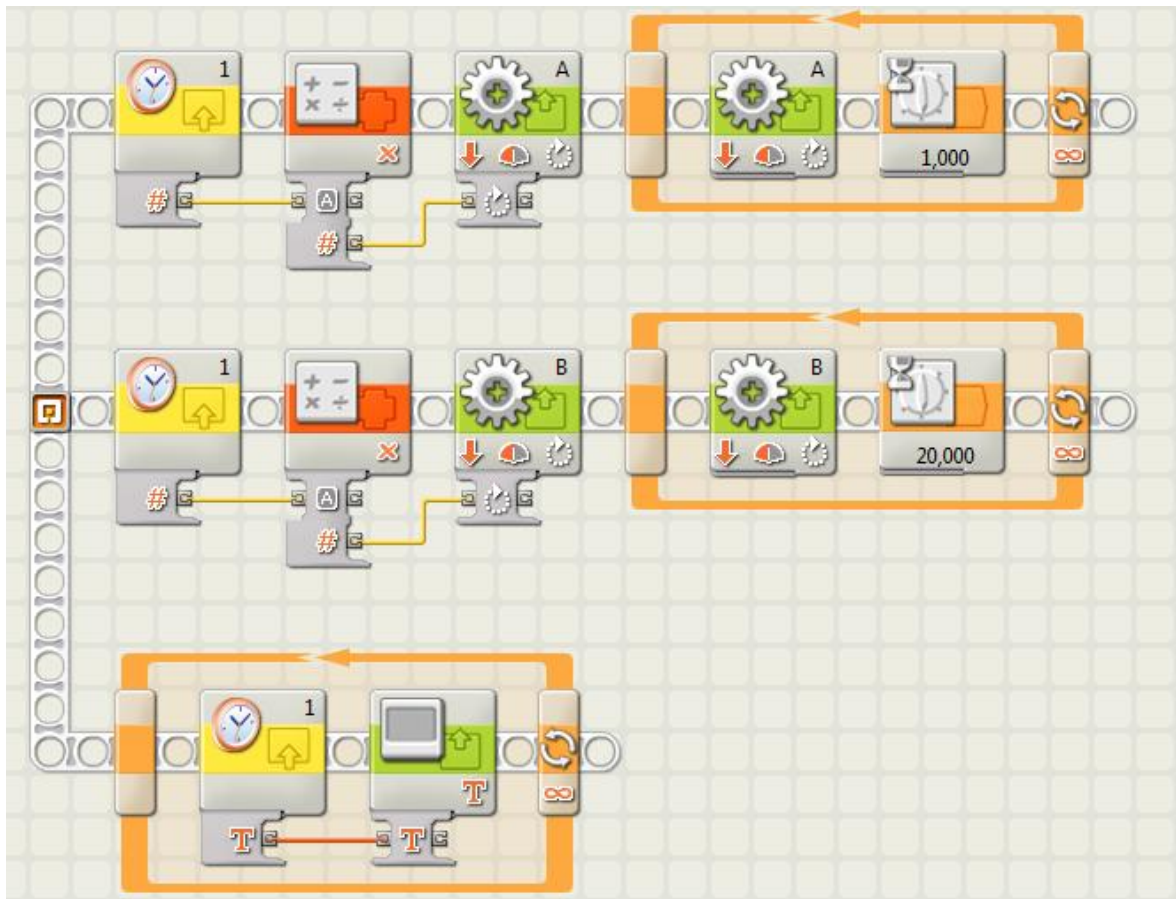
Proua Salme köögikell kukkus öösel põrandale ja läks katki. Tema naabrimees Valdur otsustas vanaprouale appi minna ning ehitas talle LEGO robotit kasutades uue seinakella.

Lahenduse idee

Kasutada võiks ühte või mitut mootorit, mis seiereid edasi liigutavad. Algse õige aja määramiseks tuleks kasutada reaalaaja kella näitu.

Üks võimalik lahendusvariant

Joonisel 32 on kujutatud üks võimalik lahendusvariant. Kasutatud on kahte NXT mootorit, mis liigutavad sekundi- ja minutiseiereid. Alguses tuleb seierid asetada käsitsi kella 12 peale (otse ülesse suunatud). Programmi käitlemisel seatakse kella seierid automaatselt õigesse asendisse. Selleks korrutatakse sekundi ja minuti hetkeväärtused 6-ga (üks täispööre on 360° ja $\frac{360^\circ}{60min} = 6^\circ$) ning tulemus antakse ette vastavale mootorile. Edasi tegutsevad kellaaja määramisel mootorid. Mootorite juhtpaneelist on valitud kestvuse (inglise keeles *Duration*) välja väärtuseks kraadid (inglise keeles *Degrees*). Sekundiseier liigub 6° sekundis ning minutiseier 2° iga 20 sekundi tagant. Näidu kontrollimiseks kuvatakse kellaega digitaalselt ka ekraanile.



Joonis 32. Ülesande nr 7 üks lahendusvariant.

Lahendusfail: ylesanne7.rbt [Lisa 4].

Tekkida võivad probleemid ning nende lahendamine

Mootorid ja reaallaja kell peavad olema ühendatud õigesti pesadesse. Tuleks jälgida, et mootorid liiguksid õiges suunas. Vajadusel muuta mootorite võimsust.

Ideed ülesande muutmiseks

Lisada võiks ka tunniseieri. Kella võib ehitada ka hammasrataste peale ja kasutada ainult ühte mootorit - see vajab pikemat nokitsemist ja katsetamist.

3.8 Ülesanne 8 - Binaarkell

Tase

Raske, vajab lisateadmisi binaarkellast.

Eesmärk

Kasutaja tutvub reaalaaja kelle pistikute väärtustega ja saab aimu binaarkella olemusest.

Ülesande täitmiseks vajalik

- LEGO MINDSTORMS NXT juhtplokk
- RTC-Nx reaalaaja kell
- Ühendusjuhtmed
- Vajadusel USB juhe arvuti ning juhtploki ühendamiseks (saab kasutada ka sinihammast)
- LEGO klotsid
- Programmeerimiskeskond NXT-G (LEGO MINDSTORMS Education NXT)
- RTC-Nx plokk NXT-G keskkonnas
- Lambikesed
- Paaris või paaritu arvu tuvastamise plokk [50]
- Andurite multiplekser(id)

Ülesande püstitus

Andrus tahab programmeerida sellist kella, mille näidust teised aru ei saaks. Õnneks tunneb ta binaararve ning otsustab valmis programmeerida binaarkella, millel tuled vilguksid ja näitaksid niimoodi õiget aega.

Märkus! Käesolevas ülesandes on valmis programmeeritud binaarkella sekundinäidu kolm parempoolsemat bitti (vastavalt 1, 2, 4).

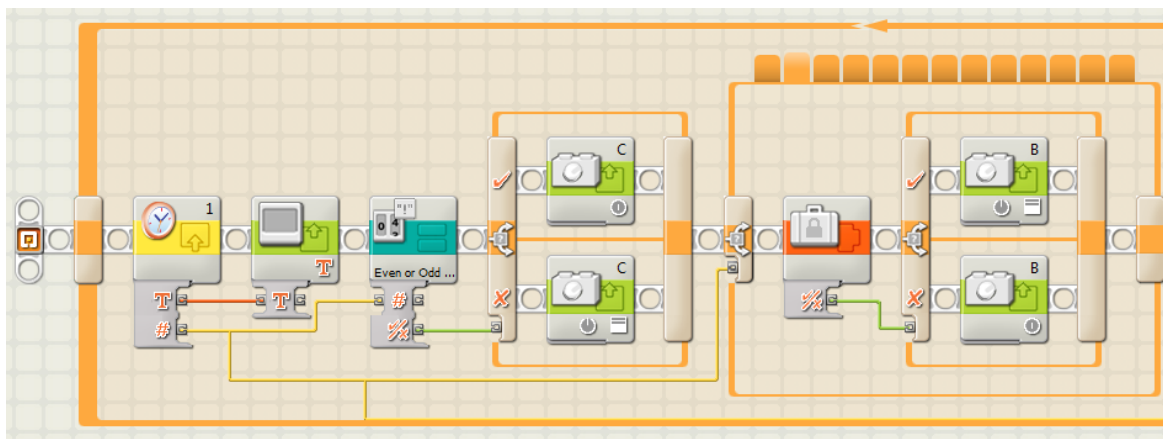
Lahenduse idee

Kasutades reaalaaja kella ning lambikesi tuleb valmis teha programm, mis kuvaks kellaaja sekundites. Näiteks kui põlevad sekundinäidu teise tulba alumine ning kolmas lambike, on väärtuseks 5. Kui kõik tuled on kustud, on väärtus 0. Binaarkellast võib lähemalt lugeda Wikipediast [51].

Üks võimalik lahendusvariant

Ülesande võimalik lahendus on näha joonisel 33. Sekundite 1. biti lambikese põlemise määramiseks on kasutatud paaris või paaritu arvu tuvastamise plokki [50]. Iga kord kui reaalaaja kella sekundivälja väärtuseks on paaritu arv, läheb alumine tuluke põlema. 2. biti

jaoks on kasutatud lüliti, mis võrdleb reaalaraja kellast saadud väärtust. Tõese väärtuse korral süüdatakse lambike ja väär puhul kustutatakse. Sarnaselt on käitunud ka järgneva lambikese jaoks. Sealkohal jääb ülesanne poolikuks kuna vaja läheks kokku 20 lambikest ja mitut multiplekseri.



Joonis 33. Osa ülesandest number 8. Joonisel on näha sekundite 1. ja 2. biti juhtimine.

Lahendusfail: ylesanne8.rbt [Lisa 4].

Tekkida võivad probleemid ning nende lahendamine

Reaalaraja kell ja lambikesed peavad olema ühendatud õigesti pesadesse. Eksimuste vältimiseks võib teha tabeli, kust on lihtne järele hoida milliste väärtuste puhul vastab lambike peab süttima.

Ideed ülesande muutmiseks

Ülesannet võiks lahendada mitmekesi ja teha valmis kogu binaarkell. Minutite ja tundide jaoks tuleb käituda sarnaselt 8. ülesandes näidatule.

Kokkuvõte

Robootikakursuste populariseerimine kooliprogrammides aitab suurendada õpilaste huvi tehnoloogia ning muidu kuivana näivate reaalainete vastu. Õppimine ei pea olema igav ja üksluine ning LEGO MINDSTORMS NXT robotite kasutamine õppetöös võimaldab seda vältida. Lisaks NXT baaskomplektile on robootikakursustes võimalik kasutada mitmeid erinevaid andureid. Üheks selliseks anduriks on reaalaja kell.

Käesoleva bakalaureusetöö eesmärgiks oli tutvuda ja kirjeldada firma Mindsensors poolt toodetud reaalaja kella (RTC-Nx-v2) DS1307 ja luua selle anduri tutvustamiseks erineva raskuastmega ülesanded. Kirjutatud materjali hakatakse kasutama Kooliroboti projekti raames õppetöös.

Lõputöö alguses kirjeldati kellaaja täpse määramise keerukust ja olulisust. Lisaks sellele tehti lühikene ülevaade aja mõõtmise ajaloo tähtsamatest sündmustest. Järgnes ülevaade andurist ning seejärel lahendamisülesanded.

Lõputöö põhirõhk oli reaalaja kellale DS1307 eestikeelse toe loomisel. Põhiliste informatsiooniallikatena kasutati anduri tootjafirma poolt loodud käsiraamatut ning anduriga töötamisel katse eksituse meetodit. Anduri tutvustamisel toodi välja selle omadused ning anti põhjalik ülevaade reaalaja kella tööpõhimõtetest. Lõputöös selgitati anduri ühendamist LEGO robotiga, selle kasutamist ning programmeerimist NXT-G programmeerimiskeskonnas. Programmeerimisülesandeid loodi mitme raskusastmetega, et rahuldada erinevate tasemetega õpilaste huvisid. Ülesannete lahendamise käigus õpivad õpilased reaalaja kella kasutama nii üksinda kui ka koostöös teiste anduritega.

Kõik bakalaureusetöö eel seatud eesmärgid said autoril täidetud. Lõputöös loodud ülesandeid on võimalik edasi arendada või soovi korral omavahel kombineerida. Lisaks sellele on võimalik teha täiendusi teiste programmeerimiskeskondade tutvustamise näol.

Real-time clock for LEGO MINDSTORMS NXT

Bachelor Thesis

Risto Soonvald

Abstract

In 2007, the project of Koolirobot was started. Its aim was to raise interest in science subjects among school students by introducing LEGO MINDSTORMS NXT robots and robotics courses to the Estonian school system.

One can use different sensors with the LEGO MINDSTORMS NXT kit. The aim of this bachelor thesis is to introduce and describe the real-time clock RTC-Nx-v2 (DS1307) sensor for LEGO MINDSTORMS NXT. In addition some exercises for lab sessions were created. The thesis consists of three chapters.

The first chapter gives an overview of the history of time. It is important for the reader to understand how our current perception of time came to be.

The second chapter focuses on the real-time clock sensor developed by Mindsensors. It gives a specific overview of the sensor's work principles. Programming introductions for the NXT-G programming software are also included in the second chapter. Furthermore, some precautions are included for those, who use the real-time sensor in extreme conditions.

The third and final chapter consists of exercises meant for robotics courses. The exercises are with various difficulties for students with different levels of experience. Each exercise has a similar layout: it contains the information and components needed to solve the exercise, a possible programming solution and gives an overview of the problems that could occur. All the solution files are on the CD that is included with this bachelor thesis.

Viited

1. Kooliroboti projekt, <http://www.robootika.ee/lego/projekt/index.php/projektist/>, (viimati vaadatud 29.05.2011)
2. The history of time, <http://library.thinkquest.org/C008179/historical/basichistory.html>, (viimati vaadatud 29.05.2011)
3. Britannica Online Encyclopedia, Sundial (timekeeping device), <http://www.britannica.com/EBchecked/media/60367/Before-precision-machine-parts-could-be-made-for-clocks-people>, (viimati vaadatud 29.05.2011)
4. Close-up: Shadow clock, <http://library.thinkquest.org/C008179/historical/shadowclock.html>, (viimati vaadatud 29.05.2011)
5. Wikipedia, Sundial, http://en.wikipedia.org/wiki/Sundial#Horizontal_sundials, (viimati vaadatud 29.05.2011)
6. Public park sundial, http://www.sundialsculptures.com/examples/public_park_sundial.jpg, (viimati vaadatud 29.05.2011)
7. Crazy Clocks: Interesting Machines That Tell Time, <http://roadtickle.com/crazy-clocks-interesting-machines-that-tell-time/>, (viimati vaadatud 29.05.2011)
8. Veekell, <http://roadtickle.com/img/strange/crazy-clocks/clepsydra.jpg>, (viimati vaadatud 29.05.2011)
9. Liivakell, <http://fathersday.indiangiftsportal.com/myshop/images/products-big/tokenz-nautic024.jpg>, (viimati vaadatud 29.05.2011)
10. Wikipedia, Galileo Galilei, http://en.wikipedia.org/wiki/Galileo_Galilei, (viimati vaadatud 29.05.2011)
11. The Biography of Galileo Galilei, <http://cnx.org/content/m11933/latest/>, (viimati vaadatud 29.05.2011)
12. Wikipedia, Pendulum, http://en.wikipedia.org/wiki/Pendulum#1656:_The_pendulum_clock, (viimati vaadatud 29.05.2011)
13. Wikipedia, John Harrison, http://en.wikipedia.org/wiki/John_Harrison, (viimati vaadatud 29.05.2011)
14. John Harrison and the longitude problem, <http://www.nmm.ac.uk/harrison>, (viimati vaadatud 29.05.2011)

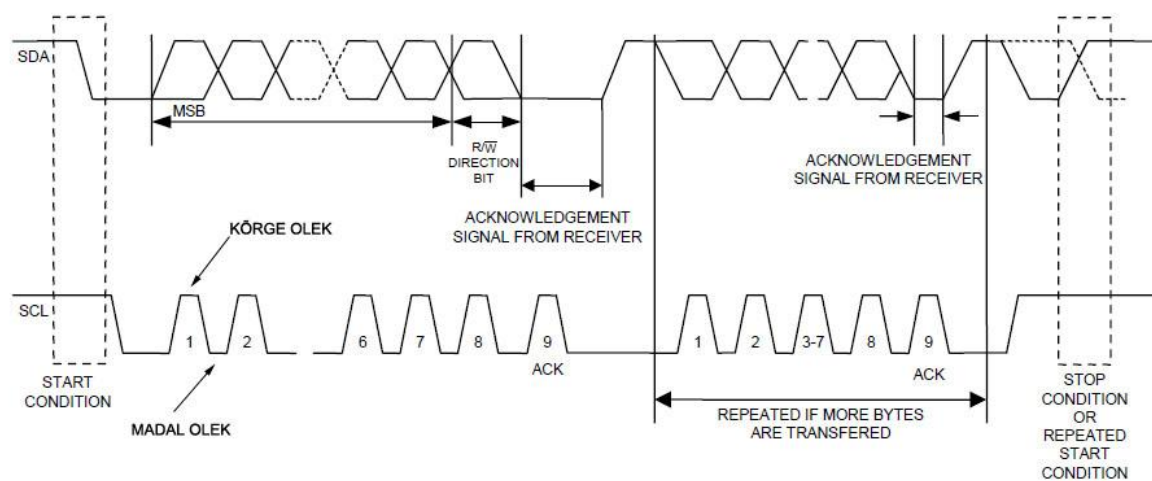
15. Wikipedia, Marine chronometer, http://en.wikipedia.org/wiki/Marine_chronometer,
(viimati vaadatud 29.05.2011)
16. John Harrisoni kell H4, <http://www.jmr.nmm.ac.uk/upload/img/D0789-1.jpg>,
(viimati vaadatud 29.05.2011)
17. Aatomkell,
http://www.sciencemuseum.org.uk/images/object_images/535x535/10319669.jpg,
(viimati vaadatud 29.05.2011)
18. Vikipeedia, Ajavöönd, <http://et.wikipedia.org/wiki/Ajav%C3%B6%C3%B6nd>,
(viimati vaadatud 29.05.2011)
19. Wikipedia, Time zone, http://en.wikipedia.org/wiki/Time_zone, (viimati vaadatud
29.05.2011)
20. Wikipedia, Greenwich Mean Time,
http://en.wikipedia.org/wiki/Greenwich_Mean_Time, (viimati vaadatud
29.05.2011)
21. Wikipedia, Royal Observatory: Greenwich,
http://en.wikipedia.org/wiki/Royal_Observatory,_Greenwich, (viimati vaadatud
29.05.2011)
22. Wikipedia, Prime Meridian, http://en.wikipedia.org/wiki/Prime_Meridian, (viimati
vaadatud 29.05.2011)
23. Wikipedia, Day, <http://en.wikipedia.org/wiki/Day>, (viimati vaadatud 29.05.2011)
24. Wikipedia, Solar time, http://en.wikipedia.org/wiki/Solar_time, (viimati vaadatud
29.05.2011)
25. Wikipedia, Coordinated Universal Time,
http://en.wikipedia.org/wiki/Coordinated_Universal_Time, (viimati vaadatud
29.05.2011)
26. Eesti Päevaleht, „Sel aastal kestab 31. Detsember kauem kui tavaliselt“,
<http://www.epl.ee/artikkel/451725>, (viimati vaadatud 29.05.2011)
27. Wikipedia, Leap second, http://en.wikipedia.org/wiki/Leap_second, (viimati
vaadatud 29.05.2011)
28. The Future of Leap Seconds, [http://www.timeanddate.com/time/leap-seconds-
future.html](http://www.timeanddate.com/time/leap-seconds-future.html), (viimati vaadatud 29.05.2011)
29. Wikipedia, Daylight saving time,
http://en.wikipedia.org/wiki/Daylight_saving_time, (viimati vaadatud 29.05.2011)

30. Vikipeedia, Suveaeg, <http://et.wikipedia.org/wiki/Suveaeg>, (viimati vaadatud 29.05.2011)
31. RTC-Nx User Guide,
http://www.mindsensors.com/index.php?module=documents&JAS_DocumentManager_op=viewDocument&JAS_Document_id=26, (viimati vaadatud 29.05.2011)
32. Realtime Clock - DS1307 Chip Datasheet,
http://www.mindsensors.com/index.php?module=documents&JAS_DocumentManager_op=viewDocument&JAS_Document_id=6, (viimati vaadatud 29.05.2011)
33. Reaalaja kell, <http://www.active-robots.com/products/mindstorms4schools/mindsensors/realtimeclock-v2/RTC-v2-400.jpg>, (viimati vaadatud 29.05.2011)
34. RJ 12 kaabel, <http://www.philohome.com/nxtplug/cable02.jpg>, (viimati vaadatud 29.05.2011)
35. E-teatmik, Binary Coded Decimal, <http://vallaste.ee/>, (viimati vaadatud 29.05.2011)
36. Wikipedia, Binary Coded Decimal, http://en.wikipedia.org/wiki/Binary-coded_decimal, (viimati vaadatud 29.05.2011)
37. E-teatmik, I²C, <http://vallaste.ee/>, (viimati vaadatud 29.05.2011)
38. Wikipedia, I²C, <http://en.wikipedia.org/wiki/I2c>, (viimati vaadatud 29.05.2011)
39. E-teatmik, Taktsagedus, <http://vallaste.ee/>, (viimati vaadatud 29.05.2011)
40. IT terministandardi sõnastik, Andmete valideerimine, <http://www.keeleeveeb.ee/>, (viimati vaadatud 29.05.2011)
41. E-teatmik, Most Significant Bit, <http://vallaste.ee/>, (viimati vaadatud 29.05.2011)
42. Vikipeedia, Kristall-ostsillaator, <http://et.wikipedia.org/wiki/Kristall-ostsillaator>, (viimati vaadatud 29.05.2011)
43. Vikipeedia, Mahtuvus, <http://et.wikipedia.org/wiki/Mahtuvus>, (viimati vaadatud 29.05.2011)
44. E-teatmik, Puhver, <http://vallaste.ee/>, (viimati vaadatud 29.05.2011)
45. Ruutlaine, <http://www.techwarereview.com/pics/square%20wave.jpg>, (viimati vaadatud 29.05.2011)
46. Wikipedia, LEGO MINDSTORMS NXT,
http://en.wikipedia.org/wiki/Lego_Mindstorms_NXT#Programming, (viimati vaadatud 29.05.2011)
47. Mindsensors kodulehekülg, <http://www.mindsensors.com/>, (viimati vaadatud 29.05.2011)

48. Reaalaja kella programmeerimisploki allalaadimine,
http://www.mindsensors.com/index.php?module=documents&JAS_DocumentManager_op=viewDocument&JAS_Document_id=42, (viimati vaadatud 29.05.2011)
49. Reaalaja kella programmeerimisploki *Help* fail
50. Paaris või paaritu arvu tuvastamise plokk, <http://us.mindstorms.lego.com/en-us/Community/NXTLog/DisplayProject.aspx?id=21029e0e-1d8b-4beb-88e1-18306a048839>, (viimati vaadatud 29.05.2011)
51. Wikipedia, Binary clock, http://en.wikipedia.org/wiki/Binary_clock, (viimati vaadatud 29.05.2011)

Lisad

Lisa 1. Andmeedastus I²C siinil



Joonis 34. Andmete edastamine I²C siinil [32].

Lisa 2. Reaalaja kella registrid

ADDRESS	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	FUNCTION	RANGE
00H	CH	10 Seconds			Seconds				Seconds	00–59
01H	0	10 Minutes			Minutes				Minutes	00–59
02H	0	12	10 Hour	10 Hour	Hours				Hours	1–12 +AM/PM 00–23
		24	PM/AM							
03H	0	0	0	0	0	DAY			Day	01–07
04H	0	0	10 Date		Date				Date	01–31
05H	0	0	0	10 Month	Month				Month	01–12
06H	10 Year				Year				Year	00–99
07H	OUT	0	0	SQWE	0	0	RS1	RS0	Control	—
08H-3FH									RAM 56 x 8	00H-FFH

Joonis 35. Reaalaja kella registrid [32].

Lisa 3. Reaalaja kella infotulp

Tabel 2. Reaalaja kella ploki infotulba selgitus [49].

	Pistik	Andme- tüüp	Väärtuste vahemik	Väärtuste tähendus
1 pesa	pesa	number	1 – 4	Pesa, millesse RTC-Nx on ühendatud
2 number 1	režiim	number	0 – 8	0 = kellaaeg, 1 = kuupäev, 2 = sekundid, 3 = minutid, 4 = tunnid, 5 = nädalapäev, 6 = kuupäev, 7 = kuu, 8 = aasta
3 number 2	tõene/väär	loogika- tehe	tõene/väär	Väljastab tõene, kui võrdlus on tõene
4 tõene/väär	kellaaeg	sõne		TT:MM:SS Kui väljastatakse „Error“ on tekkinud RTC-Nx andmete lugemisel viga
5 tõene/väär	kuupäev	sõne		PP-KK-AA Kui väljastatakse „Error“ on tekkinud RTC-Nx andmete lugemisel viga
6 number välja	sekundid	number	0 – 59	99 = viga RTC-Nx andmete lugemisel
7 number välja	minutid	number	0 – 59	99 = viga RTC-Nx andmete lugemisel
8 number välja	tunnid	number	0 – 24	99 = viga RTC-Nx andmete lugemisel
9 number	päevad	number	0 – 6	Pühapäev = 0 või 7,

välja				esmaspäev = 1, teisipäev = 2, jne 99 = viga RTC-Nx andmete lugemisel
a number välja	kuupäev	number	1 – 31	99 = viga RTC-Nx andmete lugemisel
b number välja	kuu	number	1 – 12	99 = viga RTC-Nx andmete lugemisel
c number välja	aasta	number	0 – 99	99 = viga RTC-Nx andmete lugemisel
d	võrdlemine	massiiv	0 - 255	NXT-G keskkond ei toeta seda võimalust

Lisa 4. CD ülesannete lahendusfailidega

Tabel 3. CD plaadil olevad lahendusfailid.

Faili nimi	Ülesande number ja nimi
ylesanne1.rbt	Ülesanne 1 – Anduri kasutamine kellana
ylesanne1_1.rbt	Ülesanne 1*
ylesanne2.rbt	Ülesanne 2 – Andmete kogumine
ylesanne3.rbt	Ülesanne 3 – Äratuskell
ylesanne4.rbt	Ülesanne 4 – Eestikeelne väljund
ylesanne5.rbt	Ülesanne 5 – Punamütsikese mure
ylesanne6.rbt	Ülesanne 6 – Vilkuvad tuled
ylesanne7.rbt	Ülesanne 7 – Seinakell
ylesanne8.rbt	Ülesanne 8 – Binaarkell